Diversité et système de recommandation : application à une plateforme de blogs à fort trafic

Damien Dudognon

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue par :
Damien DUDOGNON
Le 04/04/2014

Titre :
Diversité et système de recommandation : application à une plateforme de blogs à fort trafic
(convention CIFRE n°20091274)

ED MITT : Image, Information, Hypermedia

Unité de recherche :

Institut de Recherche en Informatique de Toulouse (UMR 5505)

Directeur(s) de Thèse :
Josiane MOTHE
Max CHEVALIER

Rapporteurs :

Sylvie CALABRETTO (INSA de Lyon)
Frédérique LAFOREST (Télécom Saint Étienne)

Autre(s) membre(s) du jury :

Claude CHRISMENT (Université de Toulouse) - Examinateur
Philippe LOPISTEGUY (IUT de Bayonne) - Examinateur
Gilles MONCAUBEIG (OverBlog SAS) - Invité

Diversité et système de recommandation : application à une plateforme de blogs à fort trafic

-

Convention CIFRE n° 20091274

Damien DUDOGNON

Remerciements

Je n'aurais pas pu mener à bien ce long projet sans les personnes qui m'ont soutenu tout au long de ma thèse, que ce soit mes confrères à l'IRIT, mes collègues à OverBlog, ainsi que mes amis et ma famille.

Je souhaite en premier lieu exprimer toute ma gratitude à mes directeurs de thèse, Josiane Mothe et Max Chevalier, pour la confiance qu'ils m'ont accordées tout au long de cette aventure. Leur disponibilité et leurs conseils avisés ont joué un rôle déterminant dans l'aboutissement de mon travail. Je les remercie également pour leur soutien, que ce soit dans le cadre de la thèse, comme dans un contexte plus personnel.

Je remercie mes rapportrices Sylvie Calabretto et Frédérique Laforest, ainsi que mes examinateurs Claude Chrisment et Philipppe Lopistéguy, qui ont accepté d'évaluer mon travail et de faire partie du jury.

J'adresse des remerciements tout particuliers à l'ensemble des occupants du bureau 424 que j'ai pu côtoyer : Anthony Bigot, Adrian Chifu, Claudio Gutierrez, Léa Laporte, Sébastien Laborie, Jonathan Louedec, Joël Marco, Bachelin Ralalason, Joelson Randriamparany et Bouchra Soukkarieh.

Je tiens à remercier le cercle des "rebelles" - Guillaume Cabanac, Gilles Hubert, Olivier Teste et Ronan Tournier - qui m'ont accepté à leur table chaque midi, et qui m'ont permis de découvrir leur quotidien d'enseignants-chercheurs.

Je remercie Claude Chrisment, responsable de l'équipe SIG, pour m'avoir accueilli au sein de l'équipe.

Je tiens à remercier plus généralement l'ensemble des membres de l'équipe SIG et de l'IRIT, pour leur aide et leur gentillesse au quotidien.

J'adresse également mes remerciements à Martine Labruyère de l'École Doctorale pour sa bienveillance et sa disponibilité, et pour toute l'aide qu'elle apporte quotidiennement aux doctorants.

Je souhaite également remercier David Hawking, de l'Université Nationale Australienne, et Javier Pereira, de l'Université Diego Portales, pour nos longues discussions culturelles et leurs retours sur mes travaux qui m'ont permis d'avancer.

Je n'oublie pas non plus le personnel et les étudiants de l'IUP MER, auprès desquels j'ai fait mes premiers pas dans l'enseignement.

Je remercie Julien Romanetto, Frédéric Montagnon et Gilles Moncaubeig, les fondateurs de la société OverBlog, qui m'ont intégré au sein de leur entreprise et qui ont contribué au bon déroulement de ma thèse en me fournissant des moyens techniques et humains.

Je tiens ensuite à remercier l'ensemble de la Team OverBlog qui m'a accueilli et qui m'a considéré dès le début comme un membre à part entière de l'équipe.

J'adresse plus particulièrement mes remerciements à Laurent Candillier, Antony Girault, Xavier Hausherr, Julien Houzeaux, Yannick Le Guédart et Lionel Tressens pour leur soutien technique et avec qui j'ai beaucoup appris au cours de ces quatre années. Je n'aurai jamais pu évaluer mes travaux dans des conditions réelles sans leur aide.

Je souhaite exprimer toute ma gratitude à Laurent Laborde (Keru) qui m'a accordé sa confiance et une grande liberté d'action sur l'infrastructure de la plateforme OverBlog. Sa patience, sa disponibilité, son expérience de sysadmin ainsi que les connaissances qu'il m'a transmises ont joué un rôle majeur dans la réussite de ma thèse.

Je tiens également à remercier Oleg Bartunov et Teodor Sigaev, nos experts PostgreSQL russes, pour leurs conseils avisés, ainsi que Fabrice et Marjorie qui, bien que ne faisant pas partie de la société, contribuent grandement à notre bien-être dans les locaux toulousains.

Je remercie très chaleureusement mes amis - Gaël, Jean-Charles, Johanne, Laurent, Quentin et Robert - pour leur soutien et leur présence. Je les remercie d'avoir su composer avec mes contraintes, et plus particulièrement avec mon "emploi du temps de ministre".

Je remercie également toute ma famille qui m'a toujours soutenu et encouragé.

Je remercie Audrey, qui partage ma vie depuis maintenant presque dix ans, pour sa présence et son soutien au quotidien. Je la remercie d'avoir accepté de repousser de quelques années nos projets, ainsi que les sacrifices qu'impliquaient la thèse.

Enfin, je remercie du fond du coeur ma soeur et mes parents, pour tout ce qu'ils m'ont apporté, pour les valeurs qu'ils m'ont transmises et sans qui je ne serai jamais arrivé jusque là. Je les remercie d'avoir cru en moi et de m'avoir toujours soutenu dans mes choix. Ce mémoire leur est dédié.

Ces remerciements ne sont pas exhaustifs, et je remercie tous celles et ceux qui ont contribués de près ou de loin à la réussite de ma thèse.

Résumé

Les systèmes de recommandation ont pour objectif de proposer automatiquement aux usagers des objets en relation avec leurs intérêts. Ces outils d'aide à l'accès à l'information sont de plus en plus présents sur les plateformes de contenus. Dans ce contexte, les intérêts des usagers peuvent être modélisés à partir du contenu des documents visités ou des actions réalisées (clics, commentaires, ...). Cependant, ces intérêts ne peuvent être modélisés en cas de démarrage à froid, c'est-à-dire pour un usager inconnu du système ou un nouveau document. Cette modélisation s'avère donc complexe à obtenir, et demeure parfois incomplète, conduisant à des recommandations bien souvent éloignées des intérêts réels des usagers. De plus, les approches existantes ne sont généralement pas en mesure de garantir des performances satisfaisantes sur des plateformes à fort trafic et hébergeant une volumétrie de données conséquente.

Pour tendre vers des recommandations plus pertinentes, nous proposons un modèle de système de recommandation qui construit une liste de recommandations répondant à un large spectre d'intérêts potentiels, et ce même dans un contexte où le système ne possède que peu d'informations sur l'usager. L'originalité de notre modèle est qu'il repose sur la notion de diversité. Cette diversité est obtenue en agrégeant le résultat de différentes mesures de sélection pour construire la liste de recommandations finale.

Après avoir démontré l'intérêt de notre approche en utilisant des corpus des références, ainsi qu'au travers d'une évaluation auprès d'usagers réels, nous évaluons notre modèle sur la plateforme de blogs *OverBlog*. Nous validons ainsi notre proposition dans un contexte industriel à grande échelle.

Abstract

Recommender Systems aim at automatically providing objects related to user's interests. These tools are increasingly used on content platforms to help the users to access information. In this context, user's interests can be modeled from the visited content and/or user's actions (clicks, comments, etc). However, these interests can not be modeled for an unknown user (cold start issue). Therefore, modeling is complex and recommendations are often far away from the real user's interests. In addition, existing approaches are generally not able to guarantee good performances on platforms with high traffic and which host a significant volume of data.

To obtain more relevant recommendations for each user, we propose a recommender system model that builds a list of recommendations aiming at covering a large range of interests, even when only few information about the user is available. The recommender system model we propose is based on diversity. It uses different interest measures and an aggregation function to build the final set of recommendations.

We demonstrate the interest of our approach using reference collections and through a user study. Finally, we evaluate our model on the *OverBlog* platform to validate its scalability in an industrial context.

Table des matières

Liste des tableaux

Table des figures

Introduction générale

Le Web est une source considérable de contenus qui ne cesse d'être enrichie par de nouvelles contributions des usagers. A titre d'exemple, sur la plateforme de blogs *OverBlog*[1], le nombre d'articles publiés est passé de 7,2 millions en 2010 à 12,5 millions en 2011, ce qui représente une croissance de plus de 70%.

Cette augmentation de la quantité de contenus disponibles est liée à l'évolution des pratiques : les usagers ne se contentent plus d'être consommateurs d'information ; ils sont également auteurs de contenus. Cette production de contenus va au delà de la simple création d'articles. Les internautes peuvent en effet interagir avec les autres ainsi qu'avec l'ensemble des contenus qu'ils consultent. Ces interactions se manifestent sous diverses formes, comme par l'ajout de commentaires sur les plateformes de blogs ou encore par la notation de produits sur des sites marchands.

Vis-à-vis de l'usager, l'enjeu est donc de l'aider à accéder à l'information pertinente noyée dans la masse. Il n'est pas le seul acteur à satisfaire. Les éditeurs de plateformes de blogs doivent également être considérés.

Pour ces derniers, l'enjeu est d'attirer les usagers en leur proposant notamment des contenus intéressants et qualitatifs en vue de satisfaire leur "appétit informationnel". L'objectif est de les maintenir captifs le plus longtemps possible et ainsi générer un trafic plus conséquent. En effet, le modèle économique de ces plateformes repose généralement sur cette notion de trafic (ou audience) puisque la principale source de bénéfices réside dans les revenus publicitaires. Ces revenus étant proportionnels à l'audience[2], les plateformes ont tout intérêt à la maximiser afin d'augmenter leurs revenus publicitaires.

L'objectif général des systèmes d'accès à l'information est d'établir une relation gagnant/gagnant entre l'usager et les éditeurs en permettant aux éditeurs de mettre en place des outils d'accès à l'information efficaces pour satisfaire l'usager et le fidéliser. Pour cela, deux axes ont été privilégiés par la littérature : les systèmes de recherche d'information et les systèmes de recommandation.

1. http ://www.over-blog.com/
2. http ://www.abc-netmarketing.com/Le-role-du-financement.html, Le rôle du financement publicitaire dans l'économie du web et ses facteurs explicatifs, 28 novembre 2001

Les systèmes de recherche d'information permettent à l'usager d'exprimer son besoin, sous la forme d'une requête, généralement exprimée en langage naturel. Les documents restitués sont ordonnés selon leur degré de pertinence pour la requête. Bien que les usagers soient familarisés à l'utilisation de ce type d'outil, la formulation du besoin sous forme de requête est souvent complexe et peut rendre cette requête imprécise (ambiguïté, trop généralisée ou au contraire trop spécialisée, ...).

Les systèmes de recommandation constituent une alternative intéressante vers laquelle nous avons orienté nos travaux. L'usager n'est pas sollicité : le système lui propose automatiquement des éléments potentiellement pertinents. L'effort nécessaire pour obtenir de l'information nouvelle s'en trouve réduit. Les systèmes de recommandation présentent cependant quelques inconvénients. En premier lieu, un tel outil peut être jugé comme intrusif si les éléments recommandés ne sont pas pertinents. Il est donc primordial pour le système de qualifier les intérêts réels de l'usager afin d'y répondre de manière adéquate. Différentes stratégies sont envisageables et dépendent de la possibilité ou non de qualifier l'usager. Cette qualification constitue une autre difficulté, et plus particulièrement dans le contexte de la blogosphère où nous ne disposons généralement que d'un nombre limité d'informations sur l'usager (profil).

La diversité, notion à laquelle la communauté s'intéresse de plus en plus, permet d'apporter des réponses à un certain nombre de problématiques évoquées précédemment.

La diversité vise à maximiser les chances de restituer au moins un document pertinent pour l'usager (Santos *et al.*, 2010). Elle permet également de faire émerger des besoins implicites, non nécessairement évidents mais présents, en positionnant l'usager dans un processus de découverte. Elle constitue également un moyen de faire face à la méconnaissance de l'usager et de son besoin, parfois ambigu ou imprécis.

Bien que cette notion de diversité ait été largement abordée dans le contexte des systèmes de recherche d'information (Clarke *et al.*, 2008) (Agrawal *et al.*, 2009), elle n'en est qu'à ses prémices dans le domaine des systèmes de recommandation. De plus, la communauté s'est jusqu'à présent essentiellement focalisée sur la diversité de contenu.

Cette thèse vise à apporter des solutions aux problèmes de recommandation dans le contexte de la blogosphère, et plus précisément lorsqu'aucune identification de l'usager n'est possible, en intégrant de manière originale la notion de diversité. Nous avons choisi l'axe de la recommandation afin de limiter l'engagement de l'usager pour obtenir des documents pertinents, c'est-à-dire répondant à ses

attentes, et ce tout en s'affranchissant de la formulation de requête parfois complexe et/ou approximative.

Nous proposons un modèle de système de recommandation qui, à partir du seul document visité et de la collection disponible, produit une liste de recommandations diversifiée permettant de répondre aux différents intérêts potentiels de l'usager. Cette liste est obtenue par l'agrégation de multiple mesures de sélection, chaque mesure traduisant une perception particulière de l'information et répondant par conséquent à des besoins spécifiques.

Afin de tendre vers les intérêts réels de l'usager, notre modèle intègre une phase d'apprentissage qui exploite les jugements de pertinence implicites des usagers pour identifier les intérêts qu'un document suscite. En accord avec la littérature (Joachims *et al.*, 2005) (Chapelle et Zhang, 2009), nous utilisons les clics comme jugements de pertinence.

Enfin, nous prenons également en compte une dimension souvent occultée par la littérature : les contraintes industrielles propres aux plateformes à fort trafic et le passage à l'échelle dans ces environnements contraints.

L'ensemble de nos propositions sont validées au travers de plusieurs évaluations impliquant des usagers réels ainsi que des collections de référence. La viabilité du modèle, d'un point de vue industriel et économique, est également montrée. Nous avons d'ailleurs intégré nos propositions au sein de la plateforme en ligne *OverBlog*.

Le mémoire est organisé de la manière suivante.

Une première partie regroupant les chapitres 1, 2 et 3 présente l'état de l'art en lien avec nos travaux, c'est-à-dire l'accès à l'information dans la blogosphère :

- Le **chapitre 1** définit le contexte de nos travaux, à savoir l'accès à l'information dans la blogosphère. Les particularités de cet écosystème numérique sont tout d'abord mises en exergue. Les deux principaux axes de recherche considérés par la littérature pour aider l'usager à accéder à l'information pertinente, à savoir les systèmes de recherche d'information et les systèmes de recommandation sont ensuite décrits.
- Le **chapitre 2** introduit les notions de pertinence et de similarité, puis se focalise sur la définition de la diversité qui en découle. La diversité est définie dans le cadre de la recherche d'information et dans celui des systèmes de recommandation.
- Le **chapitre 3** précise quant à lui les méthodes d'évaluation existantes et auxquelles nous faisons référence tout au long du manuscrit.

La seconde partie est consacrée à nos contributions et comporte deux chapitres :

- Le **chapitre 4** décrit notre modèle qui consiste à agréger des mesures de sé-

lection diversifiées. L'étude préliminaire validant les hypothèses fondatrices de notre proposition précède la description des différents composants du modèle. Ce dernier est ensuite évalué au travers d'une expérience utilisateur, où nous mettons également en évidence l'intérêt de la diversité pour la satisfaction de l'usager.

— Enfin le **chapitre 5** présente l'implantation du modèle au sein de la plateforme de blogs *OverBlog*, c'est-à-dire dans un contexte industriel réel soumis à un fort trafic. Après avoir défini les contraintes posées par ce contexte, ainsi que les outils mis en place pour évaluer les performances de notre modèle, nous évoquons les résultats obtenus qui démontrent la viabilité de notre approche.

Première partie

État de l'art
La Blogosphère et l'accès à l'information

Chapitre 1

Blogs et accès à l'information

1.1 Introduction

Un blog est défini par Agarwal et Liu (2008) comme un site web dynamique qui propose de manière antéchronologique des contenus appelés billets ("blog posts"). Les billets peuvent être commentés par les lecteurs, ce qui rend les blogs interactifs. Un blog est fréquemment mis à jour et alimenté par de nouveaux billets rédigés et maintenus par un auteur appelé blogueur. L'univers constitué par les blogs est appelé blogosphère.

Un blog comporte des caractéristiques particulières qui le distingue d'un autre site web, et que nous détaillons dans la section 1.2. Ces spécificités impliquent également des axes de recherche particuliers évoqués en section 1.3. Dans ce chapitre, nous détaillons ensuite les principes généraux d'accès à l'information qui s'appliquent au cas de la blogosphère, en distinguant l'accès via une requête de l'utilisateur (section 1.4) et celui par recommandation du système (section 1.5).

1.2 Caractéristiques des blogs et de la blogosphère

Les éléments qui distinguent les blogs, et par extension la blogosphère, des autres contenus web sont nombreux.

Les blogs sont en premier lieu centrés sur les individus (Mishne, 2006). Un billet se focalise sur les réactions et le ressenti des évènements par une personne plutôt que sur les évènements eux-mêmes. De plus, Mishne (2006) précise que dans la plupart des cas, il existe une relation un-à-un entre un blog et une personne donnée (l'auteur). Un blog représente donc un individu et offre un aperçu de sa vie, de son

environnement, de ses sentiments ou encore de ses centres intérêts. Dans certains cas, un blog peut être collaboratif et ainsi être l'œuvre de plusieurs auteurs.

Une seconde caractéristique fondamentale de la blogosphère est son hétérogénéité, tant au niveau de la forme qu'au niveau du fond. Ainsi, un billet peut contenir une combinaison de médias de natures diverses comme du texte, des images, des vidéos ou encore des liens vers d'autres billets, pages web ou média en rapport ou non avec le thème du billet. N'imposant aucune règle de mise en forme, le blog se veut simple et accessible. Le style d'écriture est également très libre, mêlant à la fois du langage écrit et du langage parlé, du vocabulaire issu de registres différents, ... (Mishne, 2006).

L'acte de publication est quant à lui facilité par l'utilisation de plateformes dédiées (par exemple les plateformes *Overblog*[1], *Wordpress*[2], *Tumblr*[3], ...), souvent gratuites, et qui offrent un ensemble de services permettant à l'auteur de se détacher de tout aspect technique. Il peut ainsi se concentrer sur le contenu. Les blogs constituent donc un outil d'expression accessible à tous et qui ne nécessite pas un haut niveau de compétence informatique (Agarwal et Liu, 2008).

L'interactivité est une autre composante importante de la blogosphère. Un auteur partage sa vision des faits, ses sentiments, ce qui suscite des réactions de la part de ses lecteurs, souvent des habitués (Kumar *et al.*, 2005). Ces réactions se manifestent au travers des commentaires associés au billet, auxquels l'auteur peut répondre à son tour, produisant ainsi de véritables discussions publiques. Contrairement aux wikis et aux newsgroups dans lesquels les internautes élaborent des contenus de manière collaborative, c'est-à-dire qu'ils sont tous auteurs, les billets sont généralement l'œuvre d'un seul individu (Kumar *et al.*, 2005).

La blogosphère se caractérise également par de nombreuses interconnexions. Un blog propose généralement une liste de liens hypertextes ("blogrolls") pointant vers d'autres blogs. Il est d'usage que les blogs cibles pointent également vers le blog d'origine : on parle alors de rétro-lien ("backlink"). Les étiquettes ("tags") associées aux blogs et aux billets permettent d'identifier des réseaux de blogs gravitant autour d'une thématique commune. On parle également de micro-communautés, auxquelles participent trois à vingt blogueurs, et qui, malgré leur petite taille, sont particulièrement actives (Kumar *et al.*, 2005). Du fait qu'un blog est associé à un individu, la blogosphère peut être assimilée à un réseau social (Mishne, 2006).

Enfin, la dernière caractéristique à considérer est la temporalité et la dynamique de la blogosphère. La fréquence de mise à jour, la soumission de nouveaux commentaires ou encore l'historique produit par les billets précédemment publiés

1. http ://www.overblog.com
2. http ://www.wordpress.com
3. http ://www.tumblr.com

rythment la vie du blog. Chaque blog diffuse des flux d'information XML permettant de suivre cette dynamique. Les standards de flux RSS et Atom sont les plus utilisés. D'autre part, la littérature met en évidence l'importance des aspects temporels pour un blog. En association à ces contenus personnels apparaissent de nouvelles notions permettant de les qualifier : la confiance, la réputation ou encore la popularité du blog (Tayebi *et al.*, 2007).

1.3 Axes de recherche liés à la blogosphère

Plusieurs axes de recherche visent à comprendre et analyser la blogosphère. Ainsi, Mishne (2006) identifie quatre axes liés aux particularités des blogs qu'il convient d'approfondir pour répondre aux problématiques induites par cet écosystème numérique :

— le profil des blogueurs ;
— les communautés ;
— la recherche d'information au sein des blogs ;
— la qualité des données avec notamment la détection et l'élimination du spam.

En plus des aspects évoqués précédemment, Hearst *et al.* (2008) suggèrent de s'intéresser également à la détection de tendances et d'opinions.

Cette thèse se focalise principalement sur les aspects liés à l'accès à l'information pour les visiteurs pour lesquels, contrairement aux blogueurs, il est difficile d'établir un profil du fait de leurs sessions de navigation très courtes. La notion de qualité des contenus et la lutte contre le spam sont également brièvement abordées, puisqu'elles influent directement sur la qualité des outils d'aide à l'accès à l'information.

Les plateformes d'édition de blogs mises à disposition des usagers sont bien souvent gratuites et leur modèle économique repose essentiellement sur la diffusion de publicités. Pour les sociétés éditant ces outils, le principal enjeu est de maintenir l'usager sur leur plateforme, après l'avoir capté depuis différentes sources de trafic comme les moteurs de recherche commerciaux (*Google*[1], *Bing*[2], *Yahoo*[3], ...), les réseaux sociaux (*Facebook*[4], *Twitter*[5], *Google Plus*[6], ...), ou encore depuis des liens directs sur des sites référents.

1. http ://www.google.fr
2. http ://www.bing.com
3. http ://www.yahoo.com
4. http ://www.facebook.com
5. http ://www.twitter.com
6. http ://plus.google.com

Enfin, il est primordial de mettre à disposition de l'usager, qu'il soit blogueur ou simple visiteur, des outils internes à la plateforme lui permettant d'explorer la blogosphère et de l'aider à extraire les contenus répondant à ses intérêts.

Les sections suivantes présentent les deux principaux axes de recherche envisagés dans la littérature pour aider l'usager à accéder à l'information dans la blogosphère, à savoir les systèmes de recherche d'information (section 1.4) et les systèmes de recommandation (section 1.5).

1.4 Systèmes de recherche d'information

La recherche d'information est une activité qui a pour objectif de permettre à l'usager d'obtenir des documents pertinents, c'est-à-dire en lien avec ses besoins d'information, et de lui permettre l'accès à ces documents (Lancaster, 1968).

Le fonctionnement d'un système de recherche d'information est communément schématisé par le modèle dit "en U" présenté en figure 1.

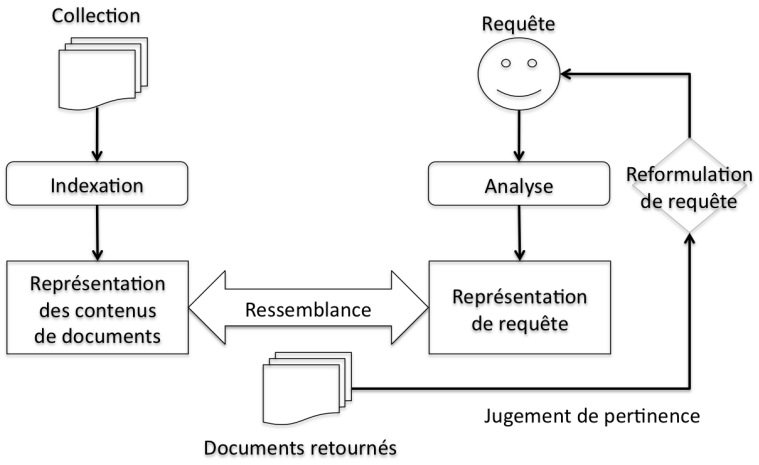


FIGURE 1 – Le modèle en U de la recherche d'information

La collection de documents disponibles, appelée corpus, est analysée lors de la phase d'indexation qui produit des représentations des documents interprétables par un système informatique. Le besoin d'information est quant à lui exprimé par l'usager lui-même au travers d'une requête en langage naturel. Cette requête est à son tour interprétée au cours d'un processus analogue à la phase d'indexation de la collection afin d'obtenir une représentation de la requête comparable à celles des documents.

La mise en correspondance du besoin de l'utilisateur avec les documents disponibles est possible grâce à des mesures d'appariement. Elles permettent au système de recherche d'information de déterminer les documents potentiellement pertinents pour l'usager. La majorité des mesures utilisées sont des mesures de similarité qui évaluent le degré de ressemblance entre les représentations des documents et la représentation de la requête.

L'évaluation par l'usager des documents proposés peut conduire à une reformulation de la requête si le besoin n'est pas satisfait ou s'il l'est de manière incomplète.

Face à la volumétrie croissante des documents [1], l'indexation manuelle, généralement réalisée par des documentalistes, a progressivement été remplacée par une indexation automatique effectuée directement par les systèmes de recherche d'information.

Les sous-sections suivantes présentent le processus d'indexation des documents et de la requête, ainsi que la mise en relation des représentations obtenues grâce aux mesures d'appariement.

1.4.1 Processus d'indexation

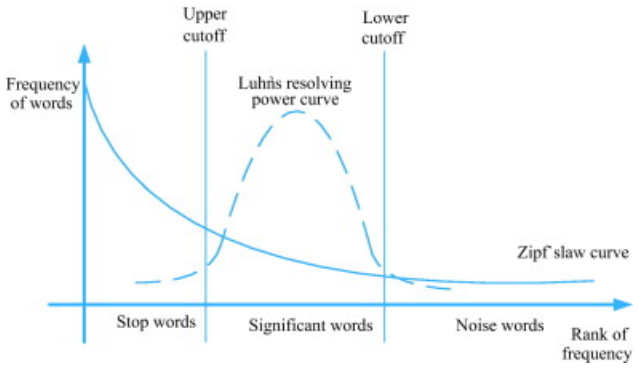


FIGURE 2 – Lois de Zipf et de Luhn (Li *et al.*, 2011)

La mise en relation de la requête et des documents disponibles est possible après en avoir produit une représentation en utilisant un langage commun. Cette phase de construction des représentations des documents et de la requête est appelée indexation. Le langage d'indexation résultant de l'analyse des documents doit offrir un compromis entre compacité et expressivité. Les principes sur lesquels repose la

1. http ://www.worldwidewebsize.com/

définition du langage d'indexation découlent des lois de Zipf (1949) et Luhn (1958), illustrées par la figure 2 et selon lesquelles :

— un terme apparaissant trop fréquemment dans un texte ne joue qu'un rôle syntaxique et ne doit pas être utilisé dans le langage d'indexation ;
— un terme présent dans l'ensemble des documents n'apporte aucun pouvoir discriminant lors de la sélection de documents ;
— un terme de fréquence moyenne est considéré comme significatif. Il représente le contenu sémantique du document et doit appartenir au langage d'indexation.

Le langage d'indexation se doit donc d'être suffisamment compact pour permettre à un système informatique de manipuler efficacement les représentations, tout en maintenant un niveau d'expressivité suffisant pour traduire aussi fidèlement que possible le contenu sémantique des documents.

L'indexation se décompose généralement en six étapes :

— **L'analyse syntaxique** : elle consiste à extraire des documents les termes ou groupes de termes servant d'unité sémantique. La ponctuation et la mise en page des documents sont ignorées. Les termes sont également mis en minuscules, voire désaccentués.
— **L'anti-dictionnaire** : l'utilisation d'un anti-dictionnaire a pour but d'éliminer les termes athématiques, ordinaires ou n'ayant qu'un rôle syntaxique. Il s'agit par exemple des pronoms, des articles, des prépositions, dont la fréquence est très importante dans les documents, mais qui ne sont pas porteurs de sens.
— **La lemmatisation** : lors de cette étape, les verbes conjugués sont remplacés par leur infinitif et les autres termes (noms, adjectifs, ...) par leur forme au masculin singulier.
— **La radicalisation** : plus agressive que la lemmatisation, elle vise à représenter les différentes formes morphologiques d'un terme par son radical. Pour la langue anglaise, l'algorithme de Porter (1980) est couramment employé. Il repose sur un ensemble de règles permettant d'obtenir les radicaux des termes. Ces règles varient d'une langue à l'autre.
— **L'analyse statistique** : elle est utilisée pour pondérer les termes d'indexation en fonction de leur pouvoir discriminant et de leur fréquence dans le document. La distribution des termes dans les documents doit également être considérée (Spärck Jones, 2003).

Chacune de ces étapes a une influence sur la qualité des représentations produites et a par conséquent un impact direct sur la qualité des résultats de la phase de recherche. Le processus d'indexation peut être adapté selon les caractéristiques du corpus (langue, type et taille des documents, ...).

1.4.2 Processus de recherche

Le processus de recherche est initié par la requête de l'usager, généralement exprimé en langage naturel ou via une suite de termes. À partir des représentations des documents du corpus et de la représentation de cette requête, obtenue par un traitement analogue à l'indexation et exprimée elle aussi à l'aide du langage d'indexation, le système de recherche d'information est capable de déterminer leur degré de ressemblance. Cette ressemblance est évaluée par l'intermédiaire de mesures d'appariement. Ces mesures reposent sur des modèles différents :

— **Le modèle booléen** : inspiré des théories ensemblistes et de l'algèbre de Boole, il s'agit d'un des premiers modèles utilisés en recherche d'information pour la fouille automatique de corpus. La présence des termes du langage d'indexation dans les représentations est déterminée de manière binaire (présence ou absence du terme), c'est-à-dire que leur poids prend soit la valeur 0 ou soit la valeur 1. La définition de requêtes utilise des opérateurs logiques ("AND", "OR", ...). Pour traiter une requête, le système recherche pour chaque terme de la requête l'ensemble des documents dont la représentation contient ce terme et réalise ensuite les opérations ensemblistes sur les ensembles de documents ainsi constitués, en conformité avec les opérateurs logiques utilisés (union des ensembles pour le "OR", intersection pour le "AND", ...). Selon ce modèle, les documents ne peuvent pas être ordonnés lors de la restitution. Une version étendue du modèle, proposée par Salton *et al.* (1983), permet l'ordonnancement des documents retrouvés.

— **Le modèle vectoriel** : proposé par Salton (1971), le modèle vectoriel représente les documents et les requêtes par des vecteurs à N dimensions, où N correspond au nombre de termes du langage d'indexation. Chacune des coordonnées du vecteur correspond au poids du terme associé. Le corpus peut être logiquement représenté par un ensemble de vecteurs ou par une matrice de dimension NxM, où M est le nombre de documents du corpus. Le principe de recherche associé consiste à représenter la requête dans le même espace vectoriel que les documents et à calculer la ressemblance de celle-ci avec chacun des documents en calculant le produit scalaire des deux vecteurs correspondants. Alternativement, la mesure Cosinus est également utilisée dans ce modèle ou toute autre mesure permettant d'estimer la similarité entre les représentations. Il est alors possible d'ordonner les documents à restituer à l'usager. Le système SMART implante ce modèle. Les modèles qualifiés de *TF.IDF* se basent sur cette approche en utilisant une pondération de chacun des termes d'indexation qui prend en compte à la fois la fréquence des termes dans le document considéré (tf et le nombre de documents de la collection contenant le terme idf).

— **Le modèle probabiliste** : le modèle probabiliste vise à produire une estimation de la probabilité de pertinence d'un document sachant une requête donnée (Robertson et Spärck Jones, 1976). Le modèle s'appuie sur la théorie des probabilités conditionnelles. Les documents restitués sont classés dans l'ordre décroissant de probabilité de pertinence. Le système OKAPI implante ce modèle ; il est basé sur une formule connue sous le nom de $BM25$ (Robertson et Spärck Jones, 1976).

De nombreux autres modèles ont été proposés dans la littérature comme les modèles à base de réseaux de neurones (Boughanem, 1992) (Mothe, 1994), les modèles reposant sur l'analyse de sémantique latente (Dumais *et al.*, 1988) ou les modèles de langue (Ponte et Croft, 1998). Cependant, ces modèles n'étant pas utilisés dans nos expérimentations, nous ne nous attarderons pas sur leur présentation.

A l'issue du processus de recherche, une liste ordonnée de documents susceptibles de satisfaire la requête est proposée à l'usager qui peut juger de leur pertinence pour son besoin.

1.4.3 Jugements de pertinence de l'usager

Les jugements de pertinence de l'usager sont soit explicites soit implicites. Dans le premier cas, l'usager indique au système les documents qui répondent effectivement à son besoin. Ce jugement peut être binaire (pertinent, non pertinent) ou peut traduire un certain degré de pertinence à l'aide par exemple d'une échelle de Likert (1932).

En l'absence de retours explicites de l'usager, il convient de déduire les jugements de ses actions vis-à-vis des documents. Il est alors possible d'inférer la pertinence des documents proposés au travers de ceux effectivement affichés par l'usager (clics), du temps nécessaire à leur lecture, de la consultation par l'usager de la liste de résultats suite à la lecture d'un premier document (rebond), ...

Les jugements implicites s'avèrent toutefois difficiles à obtenir du fait de la complexité du comportement de l'usager et des pratiques de navigation liées au Web. Par exemple, en tenant compte du temps de consultation du document, rien ne garantit que l'usager l'a effectivement lu et apprécié. En effet, le document peut avoir été ouvert en arrière plan dans un nouvel onglet du navigateur. Dans ce cas, le temps de consultation est biaisé.

Plusieurs approches se sont donc orientées vers l'analyse des clics afin de capter les retours de l'usager. Les modèles proposés diffèrent essentiellement par les principes retenus pour déduire la probabilité de pertinence d'un document à partir des clics. Ainsi, le modèle présenté par Craswell *et al.* (2008) suppose que

la probabilité de pertinence d'un document diminue avec son rang, et qu'un seul document est choisi par l'utilisateur (le premier document cliqué est considéré comme pertinent).

Guo *et al.* (2009) étendent ce modèle pour considérer des sessions, c'est-à-dire des clics multiples sur une même liste de résultats. La pertinence d'un document est alors définie comme le rapport du nombre total de clics pour le document par le nombre de sessions où il apparait dans la liste de résultats. Dans ce contexte, la pertinence est donc évaluée globalement pour l'ensemble des utilisateurs.

Au travers du modèle bayésien dynamique, Chapelle et Zhang (2009) distinguent deux niveaux de pertinence : la pertinence perçue et la pertinence effective. La pertinence perçue représente la probabilité qu'un document soit cliqué. En d'autres termes, il s'agit de la probabilité qu'un utilisateur puisse être attiré par ce document avant sa consultation. La satisfaction de l'utilisateur suite à la lecture du document est traduite par la pertinence effective.

D'autres approches considèrent qu'il existe plusieurs types de pertinence. Laporte *et al.* (2012) utilisent les clics dans le cadre de moteurs de recherche géoréférencés. Chaque élément de la liste de résultats est représenté par une fiche comportant plusieurs caractéristiques cliquables (titre, lien hypertexte, lieu, numéro de téléphone, ...). Plusieurs clics sont donc possibles pour un même résultat. Laporte *et al.* (2012) indiquent que chaque couple clic/caractéristique répond à un besoin particulier de l'usager et proposent un modèle permettant de prendre en compte cette spécificité.

Plusieurs études ont montré que l'utilisation des taux de clics comme jugements de pertinence conduisait à des résultats comparables aux jugements explicites (Joachims *et al.*, 2005) (Chapelle et Zhang, 2009).

1.4.4 Limites des systèmes de recherche d'information

Les systèmes de recherche d'information présentent un certain nombre de limites. Ils exigent tout d'abord à l'usager un certain engagement, puisque par la saisie d'une requête, il est à l'origine du processus de recherche. Sans action de la part de l'usager, aucun document n'est restitué.

L'interprétation de la requête peut également s'avérer problématique. En effet, l'expression de son besoin d'information via une requête est une tâche complexe. Il en résulte bien souvent une requête approximative, imprécise, voire ambiguë, difficilement interprétable par le système et qui ne permet pas toujours de sélectionner les documents pertinents (Spärck-Jones *et al.*, 2007).

En proposant automatiquement des documents à l'usager, les systèmes de recommandation tentent d'apporter des solutions à ce type de problèmes.

1.5 Systèmes de recommandation

1.5.1 Principes généraux

Les systèmes de recommandation sont définis comme étant “des outils logiciels et des techniques qui suggèrent aux usagers des éléments utiles” (Ricci *et al.*, 2011). Le terme général “item” est employé pour dénoter ces éléments qui peuvent être de formes très différentes : documents textuels, images, vidéos, lieux, produits commerciaux, ...

Afin d'identifier les informations à recommander, plusieurs stratégies ont été proposées dans la littérature. Elles sont généralement classées en trois catégories (Malone *et al.*, 1987) (Balabanović et Shoham, 1997) (Montaner *et al.*, 2003) (Burke, 2007) (Candillier *et al.*, 2009) :

- les approches basées sur le contenu : la pertinence des items à recommander est estimée par la similarité entre les caractéristiques ou le contenu des items, et le profil de l'usager reflétant ses besoins en termes de contenu (Magnini et Strapparava, 2001). Ce profil est établi à partir des items qu'il a déjà vus ;
- le filtrage collaboratif : le système estime la pertinence des informations en se basant sur les interactions entre les informations et les autres usagers du système (leurs jugements de pertinence). Il s'agit de recommander des items vus par d'autres usagers ayant des goûts similaires ou des intérêts communs. Ainsi, une information est considérée d'autant plus pertinente que la proportion d'usagers ayant un profil similaire et ayant apprécié cette information est élevée (Resnick *et al.*, 1994) (Breese *et al.*, 1998) (Schafer *et al.*, 2007) ;
- les systèmes hybrides : ils exploitent la complémentarité des deux approches précédentes en les combinant (Joachims *et al.*, 1997) (Wang *et al.*, 2006).

Ces trois typologies de systèmes de recommandation sont présentées plus en détail dans les sous-sections suivantes.

1.5.2 Approches basées sur le contenu

Les systèmes de recommandation basés sur le contenu tentent de proposer à l'usager des items similaires à ceux auxquels il s'est intéressé. Un profil est ainsi construit à partir de son historique pour représenter ses préférences ou ses intérêts. Le système met en relation ce profil avec les attributs des items afin de recommander de nouveaux items intéressants.

Les items sont représentés par un ensemble de caractéristiques (on parle également d'attributs ou de propriétés). Si certaines d'entre elles sont définies

à l'aide de données structurées, la plupart des caractéristiques prend la forme de données textuelles non structurées (Picot-Clémente, 2011) qui conduisent à des problématiques abordées par le domaine de la recherche d'information, comme l'extraction des termes représentatifs (discrimimants et porteurs de sens). Les solutions mises en œuvre sont comparables à celles évoquées en section 1.4.

Un exemple de système de ce type est le système de recommandation "Letizia" (Lieberman, 1995) qui s'appuie sur les actions de l'usager. Prenant la forme d'une extension d'un navigateur web, il traque le comportement de l'usager pour établir un profil alimenté à l'aide des termes relatifs aux intérêts de l'usager. Ses intérêts sont déterminés par ses jugements de pertinence implicites comme les requêtes soumises aux moteurs de recherche, les liens suivis ou encore l'ajout d'une page dans les favoris. Le profil ainsi établi est alors utilisé pour proposer à l'usager des recommandations au cours de sa navigation.

Les approches sémantiques vont plus loin que les approches basées uniquement sur des vecteurs de termes, notamment en permettant le rapprochement de concepts (Semeraro *et al.*, 2007).

Des mesures de similarité utilisées en recherche d'information sémantique (comme (Wu et Palmer, 1994), (Lin, 1998) ou (Dudognon *et al.*, 2010b)) peuvent être utilisées pour déterminer la pertinence des documents vis-à-vis d'un profil.

Une limite du filtrage basé sur le contenu est la sur-spécialisation. Ce phénomène fait référence au fait que les recommandations sont très proches d'informations déjà vues (Adomavicius et Tuzhilin, 2005).

1.5.3 Filtrage collaboratif

Les systèmes de recommandation de type filtrage collaboratif reposent sur une communauté d'usagers et sur leurs jugements de pertinence. Contrairement aux systèmes de recommandation basés sur le contenu qui ne se focalisent que sur un usager, les approches collaboratives s'appuient sur tous les usagers du système. Des systèmes de ce type sont par example : Amazon (Linden *et al.*, 2003), Netflix, Grouplens (Konstan *et al.*, 1997).

Il est maintenant bien admis que le filtrage collaboratif permet généralement de meilleures recommandations que le filtrage basé sur le contenu mais qu'il souffre d'un problème de démarrage à froid, le système étant incapable de gérer de nouveaux éléments ou de nouveaux utilisateurs (Koren *et al.*, 2009). Ainsi, le problème du traitement des nouveaux objets et celui de la nécessaire densité des appréciations ont été identifiés et des solutions y sont proposées (Sarwar *et al.*, 2000b).

1.5.4 Modèles hybrides

Les limitations de ces deux types de système de recommandation a conduit à la proposition d'approches hybrides les combinant afin de tirer partie de leurs avantages respectifs. Ces systèmes combinent les résultats des différentes techniques de recommandation (Balabanović et Shoham, 1997) (Burke, 2002). Melville *et al.* (2002) montrent que les systèmes hybrides sont particulièrement utiles lorsque peu d'information est connue sur l'utilisateur ou sur les items. Les systèmes Fab (Balabanović et Shoham, 1997) et Webwatcher (Joachims *et al.*, 1997) sont de ce type.

Les combinaisons pondérées sont probablement les modèles hybrides les plus simples. Dans les modèles hybrides pondérés, les scores des différentes composantes de recommandation sont combinées, généralement en utilisant une combinaison linéaire (Claypool *et al.*, 1999) (Burke, 2002). Les pondérations sont définies de façon empirique sur des données d'apprentissage. D'autres méthodes reposent sur une combinaison de fonctionnalités. Dans ce type de méthodes, les caractéristiques d'une source, par exemple celles d'une recommandation collaborative, sont injectées dans les caractéristiques d'un système de recommandation basé sur le contenu (Basu *et al.*, 1998).

Cependant, ces approches restent difficilement applicables dans un contexte semblable à celui de la blogosphère où l'hétérogénéité des contenus (typologies des médias publiés, qualité, format, données manquantes, ...), ainsi que les sessions de navigation des usagers très courtes, rendent la construction de profils difficile.

Chapitre 2

Diversité pour l'accès à l'information dans la blogosphère

Les outils d'aide à l'accès à l'information se donnent comme objectif de fournir aux usagers des informations "pertinentes". Par ailleurs, les outils s'appuient sur le calcul de similarité, entre la requête et les contenus pour les systèmes de recherche d'information, entre les contenus des items et/ou les profils des utilisateurs pour les systèmes de recommandation.

La section 2.1 tente de faire le lien entre pertinence et similarité et d'en indiquer la variété. Pour répondre au mieux à la variété des attentes des usagers, de nombreuses recherches s'intéressent à la diversité. La section 2.2 présente la diversité en recherche d'information ; la section 2.3 dans les systèmes de recommandation.

2.1 Pertinence et similarité

Quel que soit le moyen employé pour accéder à l'information, qu'il s'agisse d'un moteur de recherche ou d'un système de recommandation, l'objectif est de répondre aux besoins de l'usager en lui proposant des documents pertinents. Il convient donc de définir cette notion de pertinence sur laquelle reposent les différents outils. Cette notion s'avère cependant particulièrement complexe à définir compte tenu du fait qu'elle est intrinsèquement liée à un usager. Mizzaro (1998) indique en effet qu'il n'existe pas de consensus pour la définition d'une pertinence universelle, c'est-à-dire répondant à l'ensemble des intérêts de tous les usagers, mais qu'il existe un ensemble de pertinences définies selon plusieurs dimensions que sont :

— les sources d'information ;
— la représentation du problème de l'usager ;

— l'évolution du besoin dans le temps ;
— et les composants du besoin.

En accord avec cette approche, Borlund (2003) souligne le caractère dynamique de la pertinence. En effet, les intérêts de l'usager changent dans le temps en fonction de l'expérience qu'il acquiert par son vécu ou encore selon l'évolution de l'information, dans le cas par exemple d'un fait d'actualité.

Traditionnellement en recherche d'information, la pertinence est déterminée au travers de mesures d'appariement (également appelées mesures de similarité) entre les documents disponibles et un besoin exprimé par une requête. Ces mesures sont une façon d'exprimer un point de vue ou une perception particulière de l'information. La diversité des pertinences peut se traduire par une diversité des mesures permettant d'identifier des documents proches. Un score est associé à chaque document ainsi identifié pour estimer à quel point il répond au besoin. Dans la littérature, nous trouvons donc un large éventail de mesures. Nous pouvons citer par exemple :

— les mesures de similarité de contenu qui déterminent les documents similaires à partir des termes (mesure Cosinus (Salton et McGill, 1983)) ou des concepts (mesure de similarité sémantique (Wu et Palmer, 1994) (Dudognon *et al.*, 2010a)) présents dans les documents ;
— les mesures de similarité basées sur la popularité des documents telles que le BlogRank qui permet de calculer la popularité d'un blog (Kritikopoulos *et al.*, 2006) ;
— les mesures de similarité collaborative utilisées dans le filtrage collaboratif. Dans ce cas, une fonction calcule le score d'un document pour un utilisateur par agrégation des scores de ce document donnés par d'autres utilisateurs (généralement les plus similaires) (Adomavicius et Tuzhilin, 2005) ;
— les mesures de similarité organisationnelle entre documents qui traduisent le fait que des documents se retrouvent souvent ensemble classés par les usagers dans des répertoires ou des catégories (Cabanac *et al.*, 2007) ;
— les mesures de similarité navigationnelle : la similarité est identifiée au travers des chemins de navigation empruntés par les usagers (Esslimani *et al.*, 2009) ;
— les mesures de similarité sociale qui reposent sur les liens individu/individu, individu/contenu et contenu/contenu. Ces mesures, de plus en plus nombreuses du fait de l'émergence des réseaux sociaux, sont appliquées dans le cadre de réseaux d'auteurs dans (Mothe *et al.*, 2006) et (Jabeur *et al.*, 2010).

Les systèmes de recommandation reposent sur des mécanismes similaires. Les systèmes de recommandation basés sur le contenu emploient des mesures reposant sur les caractéristiques des documents, alors que le filtrage collaboratif se focalise

davantage sur la ressemblance des usagers et des traces laissées au cours de leur tâche d'accès à l'information.

Concernant les systèmes à base d'un filtrage sur le contenu, les systèmes de recommandation "Letizia" (Lieberman, 1995) et "NewT" (Sheth et Maes, 1993) utilisent par exemple la mesure Cosinus comme mesure de similarité. Ces systèmes de recommandation n'utilisent qu'une seule mesure de similarité pour produire des recommandations. Ils supposent alors que la mesure de similarité employée est à même de satisfaire les intérêts de tous les usagers. Les résultats proposés par les systèmes de recommandation construits autour d'une seule mesure sont souvent très proches. Or, comme le précisent Ziegler *et al.* (2005), des recommandations très similaires ne présentent que peu d'intérêt pour l'usager. McNee *et al.* (2006) indiquent également que la précision des recommandations n'est pas nécessairement gage de pertinence. Enfin, selon Santos *et al.* (2010), en diversifiant les résultats, la probabilité de satisfaire les intérêts de l'usager est maximisée.

Face à ce constat de pertinences multiples, et pour faire face à cette variété des intérêts, les travaux de la communauté se sont orientés vers la notion de diversité. La majorité des propositions s'est focalisée en premier lieu sur le domaine de la recherche d'information (Bradley et Smyth, 2001) (Agrawal *et al.*, 2009), l'objectif recherché étant de maximiser les chances de restituer au moins un document pertinent pour l'usager (Santos *et al.*, 2010). Ces propositions ont conduit à plusieurs définitions essentiellement liées aux besoins de l'usager.

Bien que plus tardif, l'intérêt de la communauté pour la diversité s'est également manifesté dans le cadre de la recommandation, avec des approches souvent dérivées de celles employées en recherche d'information.

C'est pourquoi nous nous focalisons d'abord dans les sous-sections suivantes sur la définition de la diversité en recherche d'information, en évoquant quelques propositions de la littérature. Nous nous intéressons ensuite à leur application pour la recommandation.

2.2 Diversité et recherche d'information

En recherche d'information, la littérature (Clarke *et al.*, 2008) (Radlinski *et al.*, 2009) distingue deux formes de diversité (cf. figure 3) : la diversité intrinsèque et la diversité extrinsèque.

La diversité intrinsèque, également nommée "nouveauté" (Clarke *et al.*, 2008), vise à proposer à l'usager, dont le besoin est clairement identifié, un ensemble de documents limitant la redondance de l'information. Elle est affectée par la connaissance que l'usager possède sur le sujet (Xu et Yin, 2008). Cette notion

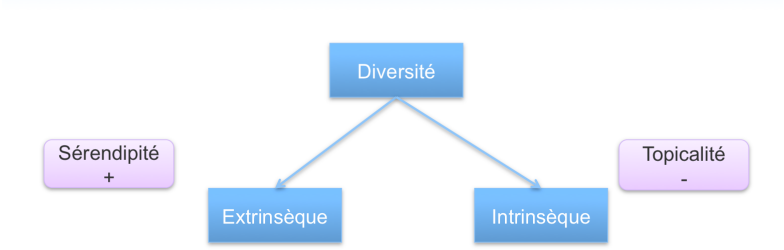


FIGURE 3 – La notion de diversité en Recherche d'Information

s'oppose à la précision, parfois appelé "topicalité" (Xu et Chen, 2006), c'est-à-dire à des résultats très similaires à la requête. Radlinski *et al.* (2009) ajoutent que ce type de diversité peut être nécessaire lorsque :

- un document seul ne permet pas de répondre complètement au besoin d'information ;
- l'usager souhaite plusieurs points de vue ;
- l'usager souhaite une sélection d'options parmi lesquelles choisir ;
- le besoin d'information est d'obtenir un aperçu du sujet ;
- plusieurs résultats provenant de différentes sources sont nécessaires pour renforcer la validité d'une réponse pour le besoin d'information.

La diversité extrinsèque s'applique quant à elle lorsque le besoin de l'usager est incertain ou ambigu. Cette incertitude peut provenir de l'ambiguïté des entités auxquelles la requête fait référence ou de l'usager lui-même. Par exemple, pour la requête "orange", il est difficile pour le système de savoir si l'usager souhaite des documents concernant le fruit, la couleur ou la société. Dans ce cas, il convient de lui proposer des documents couvrant un large panel d'intérêts potentiels (Radlinski *et al.*, 2009).

La "sérendipité" (Toms, 2000) est une alternative permettant d'obtenir ou d'améliorer la diversité. Elle consiste à proposer à l'usager des documents attractifs bien qu'inattendus, et qu'il n'aurait pas obtenus autrement (Herlocker *et al.*, 2004).

Bien que nombreuses, les approches proposées pour diversifier les résultats se focalisent généralement sur la phase de réordonnancement des résultats et il est possible de les regrouper en deux catégories :

- un premier ensemble d'approches peut être assimilé à un problème de classification, où l'on cherche à extraire des groupes correspondant aux divers intérêts exprimés par les documents ;
- le second type d'approches s'apparente à une méthode de sélection des documents comme l'approche MMR ("Maximal Marginal Relevance")

proposée par Carbonell et Goldstein (1998).

Concernant les propositions reposant sur la classification, He *et al.* (2009) utilisent un algorithme de classification "simple passe" ("Single Pass Clustering"). Le premier document de la liste de résultats initiale est sélectionné et affecté à un premier groupe. L'algorithme traite ensuite de manière séquentielle les autres résultats. Chaque document est associé au groupe duquel il est le plus proche. Si la distance document-groupe est au dessus d'un certain seuil, le document est affecté à un nouveau groupe. Bi *et al.* (2009) obtiennent de meilleurs résultats en utilisant l'algorithme des K-Moyennes (MacQueen, 1967).

Quel que soit l'algorithme utilisé, l'affectation aux différents groupes s'effectue généralement à l'aide d'une mesure de distance, comme la distance euclidienne ou la mesure Cosinus. Une pondération est parfois employée, en utilisant par exemple la fréquence des termes présents dans les documents.

He *et al.* (2009) appliquent quant à eux une méthode de classification hiérarchique combinée à un modèle de langage aux cinquante premiers documents de la liste de résultats. Des indicateurs de qualité et de stabilité des groupes guident la phase d'affectation au groupe. Le meilleur résultat de chaque groupe est sélectionné.

Dans toutes ces propositions, l'étape de classification intervient après la restitution d'un premier ensemble de documents, qui sont alors regroupés selon les intérêts identifiés par les regroupements réalisés. Elles posent l'hypothèse qu'un groupe représente un intérêt thématique particulier.

Le second type d'approche se focalise sur des méthodes de sélection des documents qui dérive généralement de l'approche MMR (Carbonell et Goldstein, 1998). On parle également de "fenêtre glissante" (Järvelin et Kekäläinen, 2002). L'approche MMR vise à sélectionner des documents qui maximisent la similarité avec la requête de l'usager, et dans le même temps, qui minimisent la similarité avec les documents déjà sélectionnés, afin de diversifier la liste de résultats finale.

La mesure employée pour évaluer le degré de similarité entre un nouveau document et les documents précédemment retenus peut différer de celle utilisée pour estimer la pertinence de ce document pour la requête. Ainsi, Kaptein *et al.* (2009) proposent par exemple deux indicateurs pour sélectionner les documents :

— le premier repose sur le nombre de nouveaux termes qu'un document apporte à l'ensemble déjà sélectionné ;
— le second considère quant à lui les nouveaux liens hypertextes entrant et sortant de ce document.

Les approches précédemment citées reposent pour la plupart sur le contenu des documents. Cependant, certains besoins de l'usager ne peuvent être traduits, et donc satisfaits, par une relation thématique. C'est dans ce cadre que la

“sérendipité” est employée. Plusieurs approches proposent donc des alternatives aux mesures de similarité reposant sur le contenu. Cabanac *et al.* (2007) proposent par exemple de considérer la similarité organisationnelle des documents qui repose sur l'analyse de leur classification au sein de dossiers par les usagers.

Les approches utilisées dans le contexte de la recherche d'information sont en grande partie transposables au domaine des systèmes de recommandation. Certaines spécificités propres à la recommandation doivent cependant être prise en compte.

2.3 Diversité et systèmes de recommandation

Depuis le début des années 2000, la diversité est devenue un challenge dans le domaine des systèmes de recommandation (Bradley et Smyth, 2001). L'objectif de la diversité dans les systèmes de recommandation est double : d'une part réduire la redondance dans la liste de recommandations et d'autre part tenir compte des divers intérêts des usagers, souvent peu ou mal spécifiés. Nous sommes par conséquent confrontés à des problématiques analogues à celles rencontrées en recherche d'information. L'idée, dans le domaine des systèmes de recommandation, est donc de maximiser l'usage de l'ensemble des informations disponibles et non de se limiter à recommander les informations “les plus populaires” ou les “plus similaires”.

Pour répondre à cet objectif, la littérature (Adomavicius et Kwon, 2012) fait mention d'une dimension supplémentaire en distinguant :

— la diversité individuelle, qui considère chaque usager indépendamment ;
— de la diversité agrégée s'appliquant à l'ensemble des usagers du système.

Par ailleurs, la définition des notions de diversité et de nouveauté diffère des définitions utilisées en recherche d'information, et revêt une dimension temporelle. La diversité peut se traduire par le fait que les éléments recommandés sont différents les uns par rapport aux autres, au moment où la liste de recommandations est présentée à l'usager. La nouveauté traduit quant à elle le fait que les recommandations apportent des éléments que l'utilisateur n'a pas déjà vus par le passé. Ainsi, comme le soulignent Vargas et Castells (2011), cette notion de diversité est étroitement liée à la notion de nouveauté.

Plusieurs approches se sont intéressées à introduire de la diversité dans les recommandations. Par exemple, Ziegler *et al.* (2005) définissent une mesure de similarité intra-liste (entre les différentes informations recommandées) sur la base d'une diversité thématique uniquement. Leur proposition est très proche de l'approche MMR.

Jabeur *et al.* (2010) proposent un modèle qui combine une mesure de similarité

de contenu et une mesure de similarité sociale. Le problème posé par ce type d'approche réside dans la manière de combiner les mesures de similarité. Ces combinaisons, qu'elles soient effectuées sous forme de combinaisons linéaires ou qu'elles soient appliquées successivement, reviennent à attribuer une certaine importance à chacune des mesures.

Une alternative à la combinaison est d'employer indépendamment différentes mesures de similarité. Le site marchand *Amazon*[1] propose par exemple plusieurs listes de recommandations à l'usager et indique le type de mesure utilisée en nommant ces listes (par exemple "Les clients ayant regardé cet article ont également regardé", "Inspirés par les tendances générales de vos achats", ...). Néanmoins, cette indépendance des mesures conduit parfois à une redondance de l'information.

Les approches de fusion offrent un moyen de solutionner ce problème. En effet, la fusion de résultats provenant de divers mesures de similarité permet d'éliminer les éventuels doublons. Schafer *et al.* (2002) et Jahrer *et al.* (2010) proposent de fusionner plusieurs sources de recommandations et présentent un "Méta système de recommandation". Selon les approches de fusion employées, il est possible de favoriser ou non les documents apparaissant dans plusieurs listes (Vogt et Cottrell, 1999).

Enfin, les approches basées sur les graphes peuvent également être utilisées pour fusionner un ensemble de mesures de similarité (Chevalier *et al.*, 2011). Le résultat de chaque mesure de similarité conduit à la définition de nouveaux liens entre les documents. Dans le graphe, les documents sont représentés par des nœuds, et les liens entre ces documents sont matérialisés par des arcs, éventuellement pondérés par les scores de similarité. Le nombre maximal d'arcs entre deux documents est défini par le nombre de mesures de similarité utilisées.

1. http ://www.amazon.com

Chapitre 3

Evaluation

Le domaine attache une grande importance à l'évaluation depuis ses débuts. Cleverdon a posé les principes de l'évaluation des systèmes de recherche d'information (Cleverdon et Kean, 1968) en termes de performances qualitatives toujours utilisés aujourd'hui. Dans le cadre des systèmes opérationnels, d'autres critères de performances se sont rajoutés. Cette section présente ces éléments en se focalisant sur ceux utilisés dans cette thèse.

3.1 Principes de l'évaluation des systèmes d'accès à l'information

C'est Cleverdon dans le projet Cranfield (Cleverdon et Kean, 1968) qui a posé les bases de l'évaluation moderne des systèmes de recherche d'information. Il a ainsi proposé l'utilisation d'une collection de tests composée d'un ensemble de documents fixe, un ensemble de besoins d'information ou de requêtes et les jugements de pertinence exhaustifs binaires (un document est pertinent ou non pour un besoin d'information) et produits manuellement.

A partir d'une collection ainsi composée, l'évaluation d'un système est réalisée de la manière suivante : le système exécute les requêtes sur la collection de documents et renvoie pour chacune une liste ordonnée de documents qu'il considère comme potentiellement pertinents. Chaque liste est ensuite comparée aux jugements de pertinence et des mesures d'efficacité sont calculées (Pinel-Sauvagnat et Mothe, 2013).

Les premières collections de test comprenaient quelques milliers de documents. Au début des années 1990, le projet d'évaluation grande échelle TREC (*Text Retrieval Conference*) débute sponsorisé par le NIST (*National Institute of*

Standards and Technology) et le département de la défense américain. TREC se donne comme ambition de fournir une base commune en termes de collection et de méthodologie d'évaluation et de démontrer la capacité des systèmes à fonctionner dans des environnements réels (Harman, 1993). Les premières collections de TREC comprennent plusieurs milliers de documents. La dernière collection distribuée par ce programme en 2009 (ClueWeb09) contient un milliard de pages web dans différentes langues, correspondant à vingt cinq téra-octets décompressés (Clarke *et al.*, 2009). D'autres collections existent dans le domaine, développées en lien avec les projets CLEF, NTCIR, ...

Compte tenu de la taille des collections de documents l'évaluation de la pertinence n'est plus réalisée de façon exhaustive mais par un principe de sondage : plusieurs moteurs de recherche renvoient un ensemble de documents qu'ils considèrent comme pertinents et ce sont ces seuls documents qui sont jugés et constituent les documents pertinents de référence.

Les moteurs de recherche commerciaux utilisent également les informations dont ils disposent pour évaluer leurs algorithmes. Dans ce cadre, il est en effet possible d'utiliser les retours implicites des usagers via leurs clics plutôt qu'un jugement de pertinence explicite (Craswell *et al.*, 2008) (Guo *et al.*, 2009). Ce point est détaillé dans la section 1.4.3.

3.2 Mesures d'évaluation qualitatives

Trec_eval [1] est l'outil utilisé dans les campagnes d'évaluation pour mesurer la performance en termes qualitatifs des systèmes. Il permet de calculer de nombreuses mesures sur les requêtes individuelles ou sur un ensemble de requêtes.

Nous ne détaillons ici que celles utilisées dans ce mémoire : les courbes rappel / précision, la précision moyenne (*Mean Average Precision* ou *MAP*) et le α-*nDCG*. Ces mesures sont largement utilisées dans la littérature, ce qui a justifié leur choix.

3.2.1 Rappel, précision et courbes rappel précision

Le rappel mesure la proportion de documents pertinents restitués et la précision mesure la proportion de documents pertinents dans l'ensemble de documents restitués. Pour une requête q donnée, ces mesures sont définies par :

$$Rappel(q) = \frac{RP(q)}{P(q)}$$

1. http ://trec.nist.gov/trec_eval/

$$Precision(q) = \frac{RP(q)}{R(q)}$$

avec :

— $RP(q)$ le nombre de documents restitués et pertinents pour q ;
— $P(q)$ (respectivement $R(q)$) le nombre de documents pertinents (respectivement restitués) pour q.

Les valeurs de ces mesures sont comprises entre 0 et 1 et sont optimales pour 1. Ces deux mesures varient en sens inverse ; aussi il est commun de montrer l'évolution de la précision à différents niveaux de rappel donnés de 0 à 1 par pas de 0,1. La précision pour une valeur de rappel r donné est alors interpolée par :

$$Prec_r(q) = \max\left(Prec_m(q)\right) \quad quel\ que\ soit \quad m > r$$

r variant de 0 à 1 par pas de 0,1.

3.2.2 Précision moyenne

La précision moyenne (*Average Precision* ou AP) est définie pour une requête q donnée par :

$$AP(q) = \frac{\sum_{r=1}^{R(q)} \left[P@r \ .rel\left(r\right)\right]}{P\left(q\right)}$$

où :

— $P(q)$ est le nombre de documents pertinents pour la requête q ;
— $R(q)$ le nombre de documents restitués ;
— r le rang ;
— et $P@r$ la précision lorsque les r premiers documents retrouvés sont considérés.

$rel\left(r\right)$ vaut 1 si le document au rang r est pertinent et 0 sinon.

La moyenne des précisions moyennes (*Mean Average Precision* ou MAP) est la moyenne arithmétique des précisions moyennes sur l'ensemble des requêtes considérées.

$$MAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q}$$

avec Q le nombre de requêtes évaluées.

Ces mesures considèrent deux niveaux de pertinence : un document est soit pertinent, soit non pertinent pour une requête donnée.

3.2.3 α-nDCG

Le *α-nDCG* (*Normalized Discounted Cumulative Gain*) proposé dans (Clarke *et al.*, 2008) dérive du *DCG* (*Discounted Cumulative Gain*) proposé dans (Järvelin et Kekäläinen, 2002) et qui permet, au contraire des mesures précédentes, de considérer plus de deux niveaux de pertinence. Il s'agit de la mesure retenue pour évaluer la tâche *diversité* de TREC Web 2009.

Le gain G est défini par :

$$G[k] = \sum_{i=1}^{m} J(d_k, i)(1-\alpha)^{r_{i,k-1}}$$

où :
- d_k le document de rank k ;
- $r_{i,k-1}$ le nombre de documents positionné avant le rank $k-1$;
- J une fonction indiquant la pertinence du document.

DCG est alors déterminé par les formules suivantes :

$$CG[k] = \sum_{j=1}^{k} G[j]$$

$$DCG[k] = \sum_{j=1}^{k} \frac{G[j]}{log_2(1+j)}$$

Le *nDCG* est le ratio entre le *DCG* et le *DCG′* (DCG idéal) :

$$nDCG[k] = \frac{DCG[k]}{DCG'[k]}$$

Généralement, α vaut $0,5$ comme proposé dans (Clarke *et al.*, 2008).

3.3 Autres mesures d'évaluation de la performance

D'autres mesures doivent être utilisées dès lors que le moteur de recherche est utilisé dans un cadre réel. Il s'agit en particulier du temps de réponse. Au-delà du temps de réponse moyen par requête, il est parfois intéressant de connaître également le taux de requêtes traitées dans un intervalle de temps donné ; la majeure proportion de requêtes devant être traitée en un minimum de temps. Certaines études indiquent en effet qu'au-delà d'un temps de chargement d'une page de trois secondes, l'utilisateur est susceptible de changer de moteur de recherche (Brutlag *et al.*, 2008).

Enfin, le taux de clics est une mesure importante dans les moteurs de recherche commerciaux dans la mesure où le modèle économique sous-jacent est lié à un taux de clics, par exemple le taux de clics sur des publicités associées aux informations renvoyées à l'utilisateur pour sa requête.

Deuxième partie

Contributions
Les systèmes de recommandation et la notion de diversité

Chapitre 4

Un modèle de système de recommandation favorisant la diversité

4.1 Vers une diversification basée sur l'agrégation des recommandations issues de plusieurs méthodes

Face à l'hétérogénéité des contenus produits par les usagers sur le Web, et plus particulièrement sur les plateformes de blogs et face à la diversité des usagers et de leurs besoins, les modèles de recommandation doivent s'adapter pour satisfaire au mieux les attentes de leurs utilisateurs.

Les systèmes de filtrage basés sur les contenus fonctionnent bien pour la recommandation de produits sur des boutiques en ligne, mais sont difficilement applicables dans le cas de contenus textuels riches et hétérogènes. C'est le cas notamment des plateformes de blogs où les contenus, générés par les usagers eux-mêmes, sont très hétérogènes. Le manque de structure commune, la forte variabilité des caractéristiques en termes de longueur ou de richesse sémantique sont autant de freins à une approche unique.

Par ailleurs, le filtrage collaboratif conduit souvent à une similarité de l'information : les recommandations sont bien souvent toutes semblables et cette redondance excessive a un intérêt limité pour l'usager.

De plus, les systèmes de recommandation collaboratifs sont difficiles à utiliser dans le contexte qui nous intéresse : la méconnaissance des usagers, liée à des sessions très courtes et au fait qu'ils ne sont généralement pas identifiés, rend la

construction de profils des usagers difficile et/ou imprécise.

Dans ce contexte, notre proposition se matérialise par un modèle de système de recommandation favorisant la diversité par l'utilisation conjointe d'une diversité de mesures de sélection. Cette diversité des mesures et des recommandations répond à un double objectif :

— améliorer la satisfaction de l'usager en lui proposant des contenus en adéquation avec ses intérêts, mais suffisamment diversifiés pour susciter chez lui de nouveaux intérêts, tout en limitant la redondance excessive des recommandations ;
— et satisfaire un plus grand nombre d'usagers en élargissant le spectre des intérêts couverts par les recommandations, et ce afin d'augmenter les chances d'obtenir au moins une recommandation pertinente pour chaque usager.

Notre proposition repose sur trois hypothèses :

Hypothèse 1 Différentes mesures de sélection, y compris lorsqu'elles s'attachent à satisfaire un objectif commun, ne produisent pas les mêmes résultats et conduisent par conséquent à des ensembles de documents pertinents différents.

Hypothèse 2 L'agrégation de différentes mesures de sélection améliore la précision.

Hypothèse 3 L'agrégation de différentes mesures de sélection a également un impact positif sur la diversité des recommandations.

Ce chapitre présente notre proposition en détail. Tout d'abord, afin de nous assurer de la pertinence de notre proposition,c'est-à-dire de l'apport d'agréger des mesures de sélection diversifiées, nous avons mené une étude préliminaire permettant de vérifier les hypothèses posées. Cette étude est présentée en section 4.2. Nous exposons ensuite une vue générale de notre modèle fondé sur ces hypothèses (section 4.3). Nous développons dans la section 4.4 les différentes mesures de sélection du modèle. Nous présentons ensuite le processus d'agrégation (section 4.5) du modèle et son modèle d'apprentissage (section 4.6). Enfin, la section 4.7 détaille l'évaluation de notre modèle au travers d'une expérience utilisateur.

4.2 Étude préliminaire

L'objectif de cette étude est de valider des hypothèses sur lesquelles repose notre proposition. Dans un premier temps, nous souhaitons montrer que des mesures de sélection différentes restituent des ensembles de documents pertinents différents,

pour ensuite montrer que l'agrégation des résultats issus de cette variété de mesures impacte positivement la précision et la diversité de l'ensemble de résultats final.

Après avoir exposé le protocole expérimental mis en oeuvre ainsi que le corpus d'évaluation utilisé, nous analysons les résultats obtenus et nous montrons qu'ils valident les trois hypothèses posées en section 4.1.

4.2.1 Protocole expérimental

Le protocole expérimental retenu est celui proposé par Lee (1997). Il consiste à comparer les listes de documents obtenues par différents systèmes pour une même requête. Le protocole repose sur l'utilisation des fichiers de résultats (*runs*) soumis par les différents participants lors de campagnes d'évaluation.

Les ensembles de résultats sont comparés deux à deux pour chacune des requêtes. Cette comparaison s'effectue à l'aide de la mesure de chevauchement (*overlap*) proposée par Lee (1997). Cette mesure permet d'évaluer la ressemblance entre deux listes de résultats. Il s'agit de calculer la proportion de documents communs parmi l'ensemble des documents retournés par les deux runs. Elle est définie comme suit :

$$overlap_k = \frac{Card(run1_k \cap run2_k) \times 2}{Card(run1_k) + Card(run2_k)}$$

où

- k est le nombre de documents considérés ;
- et $run1_k$ (respectivement $run2_k$) est le sous-ensemble de k premiers documents du $run1$ (respectivement du $run2$).

Le chevauchement *overlap* est compris entre 0 et 1. Il vaut 1 si $run1_k$ et $run2_k$ sont identiques, et 0 si $run1_k$ et $run2_k$ n'ont aucun élément en commun.

Ce chevauchement est calculé pour chaque requête indépendamment puis la moyenne non pondérée est calculée sur l'ensemble des requêtes considérées. Ce protocole présente l'avantage de pouvoir reposer sur des runs existants et donc de ne pas avoir à réimplanter les différentes approches permettant d'obtenir des runs. Cela facilite grandement la mise en œuvre de cette étude préliminaire. En effet, les runs sont généralement mis à la disposition de la communauté à l'issue des campagnes d'évaluation. Pour notre étude, nous avons choisi d'utiliser les runs de la campagne TREC Web 2009.

Le choix du protocole de Lee (1997) est également motivé par sa simplicité et sa rapidité de mise en œuvre. De plus, il nous permet d'obtenir des résultats qui pourront être comparés à ceux de Lee (1997).

4.2.2 Corpus d'évaluation

Notre étude préliminaire se base sur le corpus utilisé lors de la campagne d'évaluation TREC Web 2009 : la collection ClueWeb09 (Clarke *et al.*, 2009). Cette collection regroupe près d'un milliard de pages web extraites entre janvier et février 2009 et a été utilisée dans deux tâches différentes.

Les documents de la collection sont regroupés en deux catégories :

— la catégorie A : elle correspond à l'intégralité des vingt cinq téra-octets de données dans plusieurs langues ;

— la catégorie B : il s'agit d'un sous-ensemble de la catégorie A composé d'environ cinquante millions de documents en anglais uniquement.

Notre étude repose sur la catégorie B.

Nous nous sommes également focalisés sur deux tâches de la campagne TREC Web 2009 : la tâche *adhoc* et la tâche *diversité*.

La tâche *adhoc* a été conçue pour évaluer les performances des systèmes de recherche d'information, c'est-à-dire leur capacité à retrouver les documents pertinents pour une requête donnée. Cette tâche demande aux systèmes de restituer une liste de documents issus de la collection. Les documents sont ordonnés par ordre décroissant de pertinence supposée. Généralement, les fonctions d'ordonnancement utilisées considèrent les documents de manière indépendante et ne tiennent pas compte des documents qui apparaissent avant dans la liste de résultats.

La tâche *diversité* diffère de la tâche *adhoc* de par son processus d'évaluation. En effet, les documents ne sont plus considérés indépendamment et leur probabilité de pertinence dépend des documents les précédant dans la liste de résultats. Pour cette tâche, la liste ordonnée restituée par un système doit être constituée d'un ensemble de documents fournissant une couverture complète des sous-requêtes de la requête, c'est-à-dire des différents aspects qu'elle peut exprimer. Comme le précise Ziegler *et al.* (2005), toute redondance excessive dans la liste de résultats doit également être évitée.

Le processus d'évaluation comporte cinquante requêtes, identiques pour les deux tâches. Chaque requête comporte une description ainsi qu'un ensemble des sous-requêtes. Ces sous-requêtes ne sont utilisées que dans le cadre de la tâche *diversité*.

```
<topic number="1" type="faceted">
  <query>obama family tree</query>
  <description>
    Find information on President Barack Obama's
    family history, including genealogy, national
    origins, places and dates of birth, etc.
  </description>
```

```
  <subtopic number="1" type="nav">
    Find the TIME magazine photo essay
    "Barack Obama's Family Tree".
  </subtopic>
  <subtopic number="2" type="inf">
    Where did Barack Obama's parents and grandparents
    come from?
  </subtopic>
  <subtopic number="3" type="inf">
    Find biographical information on Barack Obama's mother.
  </subtopic>
</topic>
```

Listing 4.1 – Extrait du fichier de requêtes TREC Web 2009

Le choix de ClueWeb09 a été motivé par le fait que cette collection constitue une référence en recherche d'information de par sa taille importante et son origine (le Web). Un autre élément qui a motivé notre choix est que cette même collection a été utilisée pour deux tâches différentes. Finalement, le fait que les runs soumis par les participants sont mis à la disposition de la communauté à l'issue de la campagne d'évaluation a été un plus.

4.2.3 Approches retenues pour la comparaison

Nous avons retenu les quatre meilleurs runs évalués pour chaque tâche (*adhoc* et *diversité*) de la campagne TREC Web 2009. Cette sélection repose sur les métriques d'évaluation employées lors de cette campagne, à savoir la *MAP* pour la tâche *adhoc* et le α-*nDCG* pour la tâche *diversité*. Les approches sur lesquelles reposent les runs comparés sont présentées plus en détail dans les sous-sections suivantes.

4.2.3.1 Tâche *adhoc*

Le tableau 1 présente les résultats obtenus par les quatres meilleurs runs de la tâche *adhoc* (ceux qui ont obtenu la *MAP* la plus élevée).

Les runs reposent sur des principes très différents, ce qui nous conduit naturellement à penser que les ensembles de documents restitués pour une même requête sont différents :

— **udelIndDRSP** : ce run a été généré en utilisant le moteur de recherche *Indri*[1]. Il combine un modèle de langage avec les chaînes de Markov. Il emploie également une mesure de confiance basée sur le domaine de l'adresse

1. http ://www.lemurproject.org/indri

Id participant	Id run	MAP
UDel	udelIndDRSP	0,2202
UMD	UMHOOsd	0,2142
uogTr	uogTrdphCEwP	0,2072
EceUdel	UDWAxQEWeb	0,1999

TABLE 1 – Performances des meilleurs runs soumis à la tâche *adhoc* de TREC Web 2009

mail de l'auteur. Des listes blanches et des listes noires permettent de déterminer les domaine autorisés et ceux interdits (Chandar *et al.*, 2009) ;

— **UDWaxQEWeb** : pour ce run, la pertinence d'un document est représentée par un ensemble de contraintes. Le processus de recherche a pour objectif de trouver les documents capables de satisfaire les contraintes définies. Cette approche est complétée par une phase d'expansion de requête. Les termes d'expension liés aux termes de la requête sont extraits d'un moteur de recherche Web (Zheng et Fang, 2009) ;

— **UMHOOsd** : utilise un modèle reposant sur les chaînes de Markov. Le système de recherche d'information proposé est un système distribué construit grâce à Hadoop [1], l'implémentation Open Source de la technologie MapReduce (Lin *et al.*, 2009) ;

— **uogTrdphCEwP** : utilise la plateforme de recherche d'information Terrier [2] avec le modèle de pondération *DPH* dérivé du modèle DFR (*Divergence From Randomness*). Le processus de recherche est complété par une phase d'expansion de requête reposant sur les documents Wikipédia du corpus ClueWeb09 (McCreadie *et al.*, 2009).

4.2.3.2 Tâche *diversité*

De manière analogue à la tâche *adhoc*, les quatre runs présentés dans le tableau 2 sont ceux ayant obtenu la valeur α-*nDCG* la plus élevée à la tâche *diversité*. Ces runs reposent sur les principes suivants :

— **uwgym** : ce run joue le rôle de base de référence et ne doit pas être considéré comme un run officiel. Il a été généré en soumettant les requêtes à l'un des principaux moteurs de recherche du marché. Les résultats obtenus ont ensuite été filtrés afin de ne retenir que les documents inclus dans la

1. http ://hadoop.apache.org
2. http ://www.terrier.org

Id participant	Id run	α-nDCG@10
Waterloo	Uwgym	0,369
uogTr	uogTrDYCcsB	0,282
ICTNET	ICTNETDivR3	0,272
Amsterdam	UamsDancTFb1	0,257

TABLE 2 – Performances des meilleurs runs soumis à la tâche *diversité* de TREC Web 2009

catégorie B de la collection ClueWeb (Clarke *et al.*, 2009) ;

— **uogTrDyCcsB** : comme pour la tâche *adhoc*, ce run repose sur l'utilisation du modèle de pondération DPH DFR. Une phase d'expansion de requête est également utilisée mais elle est cette fois-ci complétée par un algorithme de classification appliqué aux documents Wikipédia restitués (McCreadie *et al.*, 2009) ;

— **ICTNETDivR3** : il s'agit de l'application de l'algorithme de classification des k-moyennes aux documents restitués lors de la tâche *adhoc*. Les documents sont affectés à la classe la plus proche en utilisant la distance euclidienne ou la mesure cosinus. Chaque groupe identifié représente une facette de la requête (Bi *et al.*, 2009) ;

— **UamsDancTFb1** : ce run repose sur une approche de type "fenêtre glissante" qui essaye de maximiser la similarité avec la requête, et en même temps, à minimiser la similarité avec les documents sélectionnés précédemment. Les documents sont sélectionnés en fonction de deux métriques : TF (*Term Filter*) et LF (*Link Filter*). TF considère le nombre de nouveaux termes uniques qu'un document apporte à l'ensemble déjà sélectionné, alors que LF utilise le nombre de nouveaux liens entrants et sortants. Le document apportant le plus d'information nouvelle (liens ou termes) est sélectionné (Kaptein *et al.*, 2009).

4.2.4 Étude de la diversité apportée par les meilleurs systèmes

Afin de montrer que deux approches différentes conduisent effectivement à des ensembles de documents différents, nous calculons la proportion d'éléments communs (*chevauchement*) pour chaque paire de runs et pour chaque requête. Cette mesure est tout d'abord appliquée à l'ensemble des runs, c'est-à-dire à des ensembles de résultats comportant 1000 documents. Nous nous intéressons ensuite

à l'évolution du chevauchement en fonction de la taille de la liste de résultats.

Les résultats présentés correspondent à la moyenne des mesures effectuées pour les quatre runs retenus. Nous avons dans un premier temps (section 4.2.4.1) calculé cette mesure en considérant les documents de manière globale, puis en nous focalisant uniquement sur les documents pertinents et les documents non-pertinents restitués. Cette démarche a été appliquée aux deux tâches de notre étude. Nous poursuivons l'analyse en étudiant l'évolution du chevauchement en fonction du nombre de documents restitués considérés (section 4.2.4.2), principalement pour nous intéresser aux premiers documents restitués. Enfin, la section 4.2.4.3 étudie la proportion de documents pertinents et de documents non pertinents en lien avec le chevauchement.

4.2.4.1 Chevauchement pour 1000 documents

Dans (Lee, 1997), le chevauchement est calculé pour les 1000 documents restitués pour une requête donnée. L'étude est menée sur le corpus de la campagne TREC 3. De la même manière, nous avons calculé le chevauchement pour les 1000 documents de chaque requête pour les tâches *adhoc* et *diversité* de la campagne TREC Web 2009. Le tableau 3 présente les résultats obtenus en ne considérant que les documents pertinents, puis uniquement les documents non pertinents.

Collection	**Documents considérés**	**Chevauchement**
TREC 3 *adhoc*	Pertinent	0,7824
	Non Pertinent	0,3519
TREC Web 2009 *adhoc*	Pertinent	0,7544
	Non Pertinent	0,4968
TREC Web 2009 *diversité*	Pertinent	0,2382
	Non Pertinent	0,0645

TABLE 3 – Chevauchement moyen en considérant 1000 documents

Nous constatons que les résultats obtenus pour la tâche TREC Web 2009 *adhoc* sont comparables à ceux obtenus lors de TREC 3 *adhoc*. Le chevauchement des documents pertinents est élevé alors que celui des non pertinents demeure limité. Ces résultats sont en accord avec les conclusions de Lee (1997) :

— les différentes approches restituent globalement les mêmes documents pertinents ;
— les documents non pertinents changent de façon importante.

Les résultats de la tâche *diversité* diffèrent de ceux obtenus avec la tâche *adhoc*. En effet, le chevauchement des documents, qu'ils soient pertinents ou non pertinents, est faible (inférieur à 0,2382). Dans ce contexte, nous constatons que les approches proposées conduisent à des documents pertinents différents, et ce pour une même requête.

Bien que les campagnes d'évaluation aient choisi de considérer des ensembles importants de documents (liste de 1000 documents pour TREC), les usagers réels se contentent généralement des premiers résultats. Afin de compléter cette étude, nous nous intéressons à présent à l'évolution du chevauchement en fonction du nombre de documents restitués (de 1 à 100 documents).

4.2.4.2 Précision vs chevauchement

Nous nous intéressons dans un premier temps à la comparaison de l'évolution de la précision moyenne et du chevauchement global (*overlap*) en fonction du nombre de documents retournés.

La figure 4 présente les résultats obtenus pour la tâche *adhoc*. Nous notons que lorsque la précision atteint sa valeur maximale (P@10, la précision pour 10 documents), le chevauchement se situe à 0,255, ce qui est relativement faible. Le chevauchement demeure faible y compris lorsqu'un nombre important de documents est restitué (P@100) puisque la limite de cette valeur tend vers 0,4.

Le phénomène est plus marqué pour la tâche *diversité* : le chevauchement est quasi nul lorsque nous ne considérons que les premiers documents (1, 2, 5 et 10) alors que la précision atteint sa valeur maximale P@2. La précision décroit de manière importante au-delà de 20 documents. Le chevauchement reste très faible (inférieur à 0,1) et ce jusqu'à 100 documents.

Les figures 4 et 5 montrent également que les têtes de liste présentent une proportion de documents pertinents plus importante que de grands ensembles de documents.

Compte tenu des résultats obtenus dans la section 4.2.4.1, nous souhaitons vérifier que, lorsque nous considérons des ensembles de résultats réduits (inférieurs à 100 documents), le chevauchement est également faible pour les documents pertinents. Dans la section suivante, nous nous intéressons donc au chevauchement des documents pertinents et des documents non pertinents.

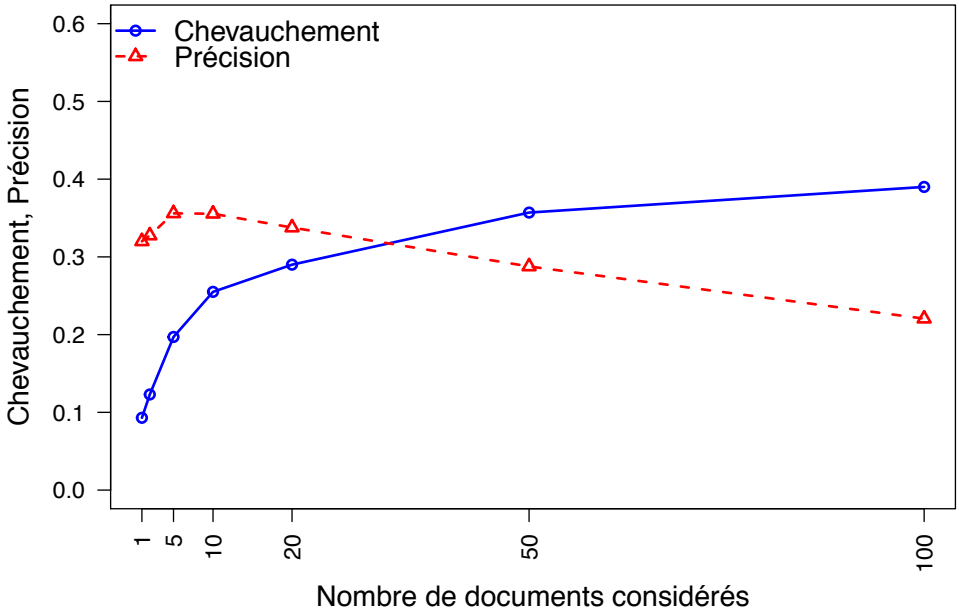


FIGURE 4 – Évolution du chevauchement et de la précision en fonction de la taille de la liste de résultats pour la tâche *adhoc* de TREC Web 2009

4.2.4.3 Chevauchement en considérant les documents pertinents vs non pertinents

Pour cette dernière partie de l'étude, nous nous focalisons sur la proportion de documents pertinents et non pertinents en fonction du nombre de documents restitués. La figure 6 montre pour la tâche *adhoc* que lorsque nous considérons un petit nombre de documents (entre 1 et 10), le chevauchement est faible (inférieur à 0,25) à la fois pour les documents pertinents et pour les documents non pertinents.

Le phénomène est amplifié dans le cas de la tâche *diversité* (figure 7), et ce même lorsqu'un nombre conséquent de documents est restitué (jusqu'à 100 documents). Le chevauchement n'excède pas la valeur de 0,15.

Ces résultats montrent que les premiers résultats restitués par différentes approches, y compris lorsqu'elles visent un objectif commun, sont différents. En ne considérant que 10 documents, la probabilité de trouver un même document dans plusieurs résultats est très faible. La première hypothèse est donc vérifiée et

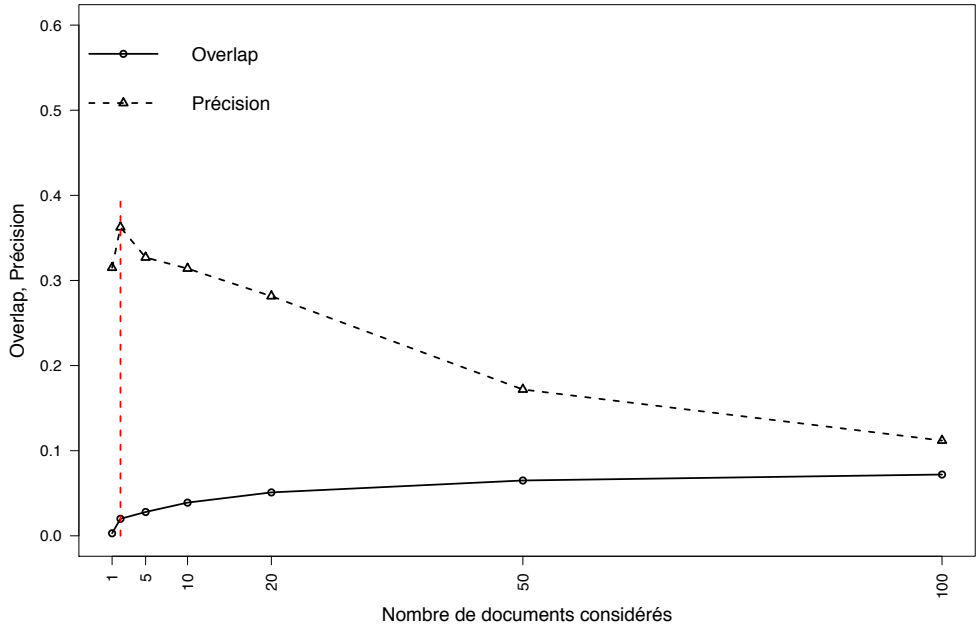


FIGURE 5 – Évolution du chevauchement et de la précision en fonction de la taille de la liste de résultats pour la tâche *diversité* de TREC Web 2009

nous amène aux deux autres hypothèses, à savoir l'intérêt pour la précision et la diversité des recommendations d'agréger des documents d'origines différentes.

4.2.4.4 Combinaison des résultats de différentes approches

Afin de confirmer les deux dernières hypothèses, nous utilisons l'approche *CombMNZ* pour fusionner les résultats issus des différents runs des tâches *adhoc* et *diversité* pour produire deux nouveaux ensembles de résultats. Les métriques d'évaluation (*MAP* et α-*nDCG*) ont alors été calculées.

Pour la tâche *adhoc*, nous avons obtenu une *MAP* égale à 0,237. Ce score dépasse celui du meilleur run soumis lors de la campagne TREC (0,2202). Dans le cas de la tâche *diversité*, nous avons obtenu une valeur α-*nDCG* égale à 0,283. Bien que ce score soit inférieur à celui du run de référence *Uwgym* (0,369), il s'avère légèrement supérieur à celui du meilleur run officiel (0,282).

Ces résultats valident donc les hypothèses 2 et 3. En effet, nous constatons que

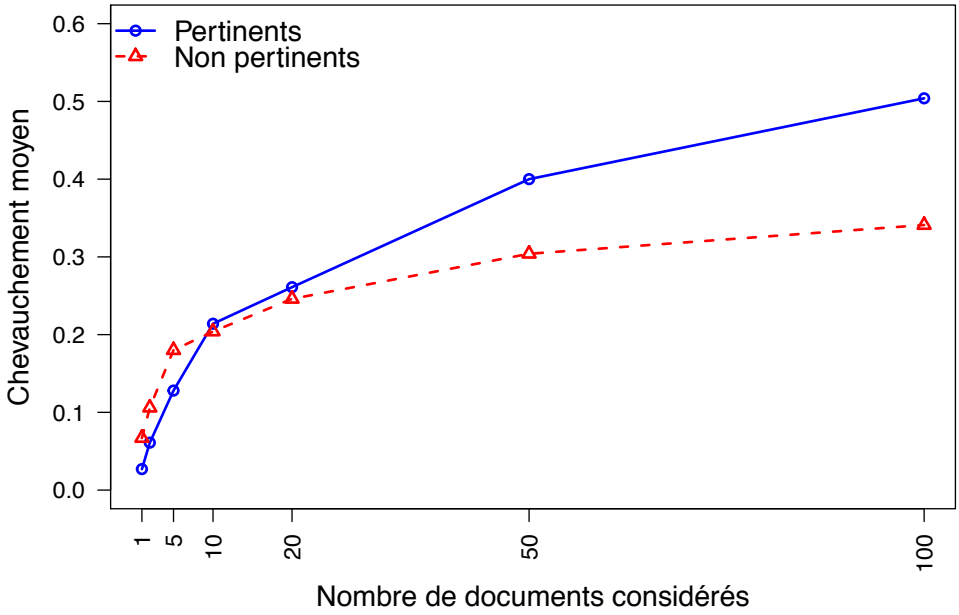


FIGURE 6 – Évolution du chevauchement des documents pertinents et non pertinents en fonction de la taille de la liste de résultats pour la tâche *adhoc* de TREC Web 2009

l'agrégation des runs améliore la précision, et impacte positivement la diversité.

4.2.5 Conclusions

Nos résultats diffèrent de ceux obtenus par Lee (1997) puisque ses résultats, obtenus par l'analyse du corpus TREC 3 (Harman, 1994) montrent que seuls le chevauchement des documents non pertinents est faible, c'est-à-dire que les différentes approches comparées retournent globalement les mêmes documents pertinents.

La taille de la collection utilisée pour les expérimentations constitue un premier élément pouvant expliquer cette différence. En effet, le corpus TREC 3 comporte environ 1 million de documents alors que ClueWeb 2009 compte environ 50 millions de documents. Cette taille plus importante conduit naturellement à une dilution des documents pertinents. La nature des documents des deux collections peuvent

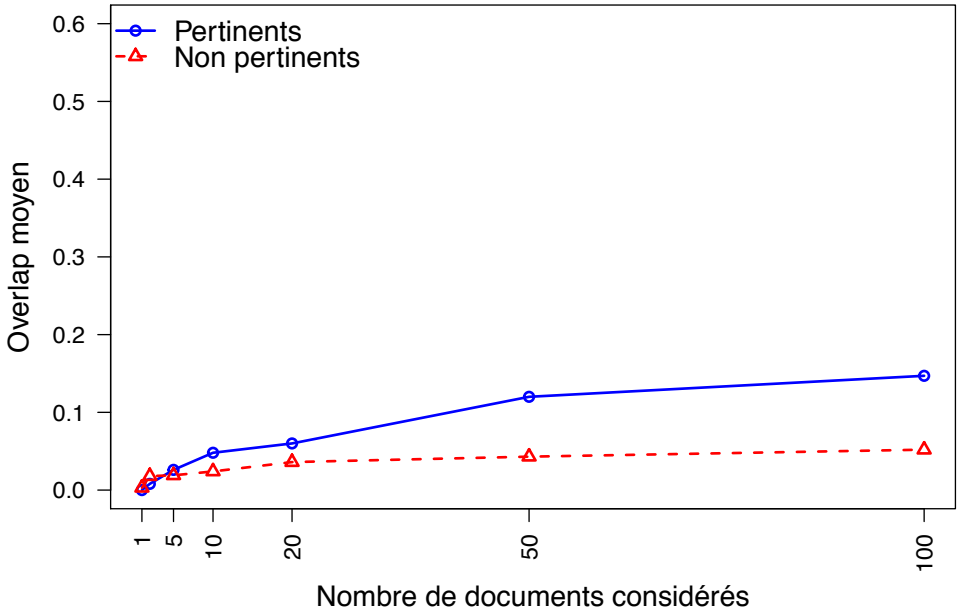


FIGURE 7 – Évolution du chevauchement des documents pertinents et non pertinents en fonction de la taille de la liste de résultats pour la tâche *diversité* de TREC Web 2009

également expliquer les différences observées. En effet, alors que la collection TREC3 comporte des documents homogènes (articles de journaux), la collection ClueWeb est constituée de documents issus du web, plus hétérogènes, mais aussi plus proches de notre contexte de travail que constituent les blogs.

Le second point que nous mettons en évidence est que l'analyse de Lee ne porte que sur des ensembles de 1000 documents. Dans notre étude, nous nous sommes intéressés à des ensembles de documents plus restreints (jusqu'à 100 documents), plus représentatifs du nombre de résultats consultés par les usagers réels. Les résultats obtenus valident les trois hypothèses formulées initialement. Ainsi, nous notons qu'une diversité d'approches, y compris lorsqu'elles visent la satisfaction d'un objectif commun (une même requête par exemple), produisent des résultats différents et conduisent par conséquent à des ensembles de documents pertinents différents.

Ces documents pertinents étant généralement les premiers documents restitués

pour chaque approche, nous pouvons considérer suite à nos expérimentations que l'agrégation des têtes de liste a un impact positif à la fois sur la précision, mais également sur la diversité des résultats.

4.3 Vue globale du modèle

L'étude menée précédemment valide les trois hypothèses fondatrices de notre modèle. En effet, nous avons montré de façon empirique que différentes approches conduisent à des ensembles de documents pertinents différents, mais également que l'agrégation des résultats qu'elles produisent améliore la précision tout en favorisant la diversité.

Le modèle que nous proposons repose donc sur l'utilisation conjointe et de manière équitable de plusieurs mesures de sélection, et ce afin de garantir la diversité dans la liste de recommandations. Nous pensons couvrir ainsi un spectre plus large d'intérêts potentiels.

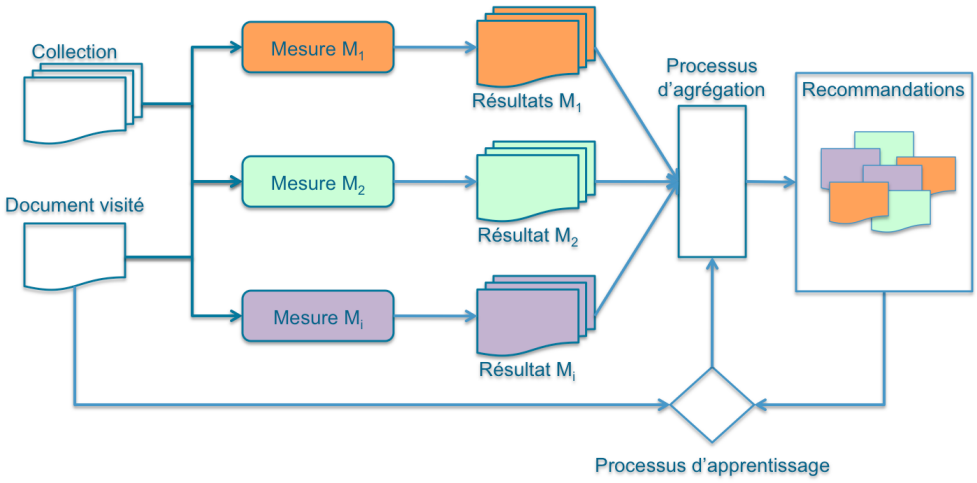


FIGURE 8 – Vue globale du modèle de système de recommandation reposant sur l'agrégation de mesures d'intérêts

Le modèle se décompose en quatre composants principaux, présentés par la figure 8 :

— les entrées du système sont constituées de la collection de documents d'où seront extraites les recommandations et le document visité ;

— un ensemble de mesures de sélection. Chacune détermine, à l'aide de scores, les documents de la collection potentiellement pertinents pour l'usager, suite à la consultation du document visité. Les documents ayant les meilleurs scores forment le résultat d'une mesure de sélection ;
— le processus d'agrégation qui combine les résultats issus des différentes mesures de sélection assurant ainsi la diversité des recommandations ;
— enfin un processus d'apprentissage analyse les jugements de pertinence (implicites ou explicites) des usagers afin de déterminer les recommandations qui s'avèrent effectivement pertinentes pour l'usager. Il permet, au cours du processus d'agrégation, d'ajuster les proportions de résultats issus de chaque mesure de sélection pour construire la liste finale de recommandations.

Nous explicitons dans les sections suivantes chaque composant.

4.4 Mesures de sélection

Cette section a pour objectif de définir la notion de mesure de sélection. Au sens où nous l'entendons, les mesures de sélection ne sont pas limitées aux seules mesures de similarité employées traditionnellement en recherche d'information. Une mesure de sélection permet d'apporter une réponse à un ou plusieurs types d'intérêts des usagers. Ces intérêts peuvent être liés au contenu des documents ("Je souhaite obtenir des documents qui abordent telle ou telle thématique"), mais également à d'autres critères comme l'origine des documents ("Je souhaite des documents du même auteur") ou sa popularité ("Je souhaite connaître les documents qui font le buzz").

Dans les sections suivantes, nous nous focalisons en premier lieu sur la définition de cette notion de mesure de sélection, pour ensuite l'illustrer par des exemples de mesures issues des expérimentations que nous avons menées sur la plateforme *OverBlog*.

4.4.1 Définition des mesures de sélection

Dans notre contexte de recommandation, une mesure de sélection restitue, à partir d'une collection et du document consulté, un ensemble ordonné de documents potentiellement pertinents.

Une mesure de sélection peut être formalisée par une fonction qui associe un document initial d_0 (le document consulté) à un ensemble de couple (d_i, w_i) où d_i est un document issu de la collection et w_i le score associé à ce document par la mesure de sélection. Elle s'exprime alors de la manière suivante :

$$f(d_0) = \{(d_i, w_i)\}$$

Une mesure de sélection considère les documents de la collection dans leur intégralité et peut exploiter toutes les caractéristiques disponibles des documents (par exemple : son contenu textuel, sa date, son auteur, ...). Ainsi, le degré de pertinence des documents restitués par la mesure d'intérêts n'est pas nécessairement limité au contenu. D'autres caractéristiques, comme les statistiques de fréquentation des documents disponibles, peuvent être considérées pour proposer des documents.

Chaque mesure est indépendante des autres et elle produit un résultat qui lui est propre. Elles peuvent donc être calculées indépendamment et de manière distribuée, répondant ainsi aux problématiques de performance. Cette indépendance des mesures contribue également à l'extensibilité du modèle :

— le nombre de mesures de sélection pouvant être intégrées au système de recommandation n'est pas limité ;
— par conséquent le modèle peut être étendu par des mesures additionnelles pour prendre en compte de nouveaux intérêts.

La section suivante décrit les différentes mesures de sélection que nous avons utilisées au cours de nos expérimentations. Les principes sur lesquels elles reposent sont particulièrement diversifiés.

4.4.2 Mesures utilisées sur la plateforme *OverBlog*

Comme précisé dans la section précédente, les mesures de sélection ne se focalisent pas nécessairement sur le contenu textuel des documents. Toutes les caractéristiques disponibles peuvent être exploitées afin d'obtenir une diversité de mesures, et par conséquent une diversité des résultats.

4.4.2.1 Caractéristiques des documents

Dans le contexte de la plateforme de blogs *OverBlog*, les documents (billets) disponibles présentent un grand nombre de caractéristiques, comme par exemple :

— le titre du billet ;
— son corps, c'est-à-dire son contenu textuel ;
— ses dates de création, de modification et de publication ;
— son auteur ;
— sa catégorie dans la taxonomie définie pour la plateforme *OverBlog* ("Cuisine", "Sport", "High Tech", ...) ;
— son URL ;
— le nom de son blog d'appartenance ;
— l'URL de ce blog ;
— la description du blog ;

— les commentaires associés au billet ;
— ou encore les statistiques du blog et du billet (nombre de pages vues, nombre de visiteurs uniques, provenance des visites, ...).

Les mesures proposées dans le cadre des expérimentations exploitent différemment ces caractéristiques afin de répondre à des intérêts différents.

4.4.2.2 Mesures de sélection utilisées

Afin de simuler les différents types de diversité, nous avons défini cinq mesures de sélection :

— **blogart** : sélectionne de manière aléatoire des billets issus du même blog que le billet initial. Ces billets sont ordonnés aléatoirement. Sur la version utilisée de la plateforme *OverBlog*, un blog n'est associé qu'à un seul auteur. Ainsi, cette mesure peut être vue comme "les billets du même auteur".
— **topcateg** : restitue les billets les plus populaires, c'est-à-dire ceux ayant généré le plus fort trafic sur une période donnée, dans la même catégorie que celle du blog d'où est extrait le billet. Les billets sont ordonnés par trafic décroissant. Ainsi, ces billets reflètent les "buzz" du moment, pour une thématique donnée.
— **search** : utilise le moteur de recherche open source *Apache Solr*[1] pour restituer des billets. La requête soumise correspond au titre du billet initial. Le moteur de recherche utilise un modèle de recherche mixant le modèle booléen et le modèle vectoriel. Un système de pondération dérivé du modèle TF-IDF (Spärck Jones, 1972) permet d'ordonner les documents. Cet ordonnancement des documents est également conditionné par leur fraîcheur : les documents les plus récents sont favorisés.
— **kmeans** : applique l'algorithme de classification des K-moyennes aux 50 premiers résultats obtenus par la mesure *search*. Le nombre de groupes est fixé arbitrairement à 10. Le document le plus proche du centre de chaque groupe est sélectionné pour construire l'ensemble de résultats final. Enfin, les documents sont ordonnés en fonction de leur poids initial (issu de la mesure *search*).
— **mlt** : repose sur le module $MoreLikeThis$ du moteur de recherche Solr. En accord avec les lois de Zipf (1949) et Luhn (1958), cette mesure extrait du document initial les termes les plus représentatifs sur la base des paramètres tf et idf. Ces termes, au nombre de 10 dans notre cas, sont ensuite utilisés pour construire une requête soumise au moteur de recherche.

Les résultats obtenus par chaque mesure de sélection répondent plus particulièrement à une facette des intérêts usager. Afin de proposer à l'usager une liste de

1. http ://lucene.apache.org/solr

recommandations répondant à un large spectre d'intérêts, notre approche agrège les résultats de chaque mesure et construit ainsi une liste diversifiée. Ce processus d'agrégation est décrit dans la section suivante.

4.5 Processus d'agrégation

Dans notre modèle, les résultats des différentes mesures de sélection sont agrégés pour produire une liste unique de recommandations. Cette étape pose plusieurs problèmes à résoudre :

- comment sélectionner les résultats provenant de chaque mesure de sélection ?
- comment garantir la diversité des résultats dans la liste finale ?
- comment agréger les résultats ?
- comment les ordonner ?
- etc...

Dans notre contexte, les attentes de l'usager sont inconnues et nous ne disposons d'aucune information nous permettant de les déduire. Par conséquent, nous ne pouvons faire aucune supposition sur ses attentes.

Le *processus d'agrégation* doit par ailleurs garantir la diversité de la liste de recommandations finale en exploitant au mieux la diversité des mesures de sélection employées et des résultats qu'elles produisent. Nous souhaitons en effet favoriser la présence de documents issus de chaque mesure de sélection dans la liste finale de recommandations.

Enfin, compte tenu de notre domaine d'application, à savoir les plateformes de contenus soumises à un fort trafic, le coût en termes de temps de calcul doit demeurer raisonnable.

Afin de répondre à ces différents problèmes, nous proposons un processus d'agrégation en trois temps :

- tout d'abord, nous procédons à une première sélection des documents issus des résultats des différentes mesures de sélection offertes par le système ;
- vient ensuite l'agrégation à proprement parler de ces différents ensembles de résultats pour produire une liste unique de recommandations ;
- enfin, les documents sont ré-ordonnés afin de mettre en avant les documents couvrant le plus grand nombre d'intérêts.

Dans les sous-sections suivantes, nous nous intéressons successivement à ces trois étapes.

4.5.1 Sélection des résultats

La première phase du processus d'agrégation consiste à sélectionner les meilleurs documents issus des différentes mesures de sélection employées.

Les mesures de sélection restituent les documents en les ordonnant par score décroissant. Par conséquent, les documents les plus susceptibles d'être pertinents sont logiquement localisés en tête de liste. Cette hypothèse a été validée par plusieurs études (Lee, 1997) (Candillier *et al.*, 2012) qui sont détaillées en section 4.2.

En nous limitant à la sélection des N premiers documents de chacune des mesures de sélection, nous maximisons les chances d'obtenir des documents pertinents. De plus, le fait de considérer pour chaque mesure un nombre restreint de documents permet d'améliorer la représentativité des différentes mesures agrégées. Le nombre de documents sélectionnés pour chacune des mesures est un des paramètres du système.

Enfin, le choix d'un nombre réduit de documents pour chaque mesure (par exemple 10 documents) a un impact direct sur les temps de traitement liés à l'étape de fusion des listes de résultats. En effet, plus les listes de résultats sont petites, moins il faut de temps pour les agréger.

4.5.2 Agrégation des listes de résultats

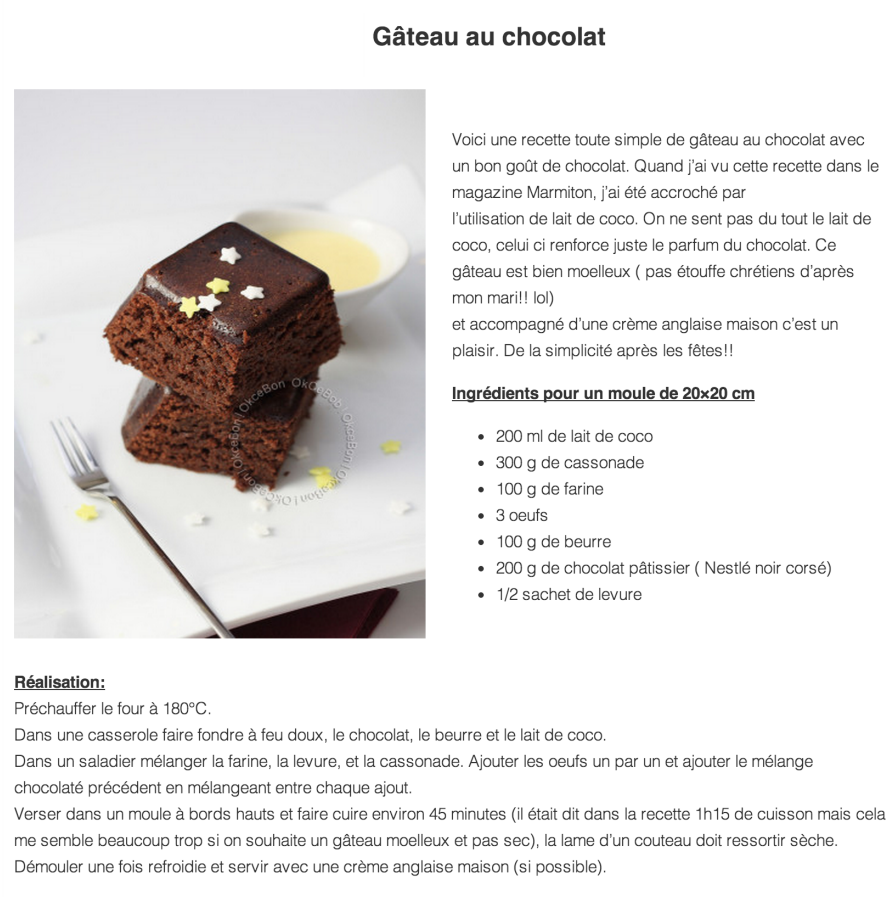
Le processus d'agrégation constitue un composant majeur de notre modèle. Il a la charge d'agréger les résultats issus des différentes mesures de similarité pour produire un ensemble visant à garantir la diversité des intérêts satisfaits.

Il existe de nombreuses façons de combiner des listes de résultats qui généralement déterminent l'importance donnée à chacune des mesures *a priori*. Par exemple il est possible de calculer le score d'un document à partir des scores qu'il a obtenus dans les différentes listes et en y appliquant une fonction comme le maximum ($CombMax$), la somme ($CombSUM$) ou encore la moyenne ($CombMNZ$) des scores (Fox et Shaw, 1994) (Hubert *et al.*, 2007). Jabeur *et al.* (2010) proposent quant à eux un modèle qui combine une mesure de contenu et une mesure sociale. Le problème posé par ce type d'approche réside dans la manière de combiner les mesures de sélection. En effet, elles tendent à favoriser les documents présents dans plusieurs listes au détriment des documents très pertinents pour une seule mesure. Ceci ne favorise donc pas nécessairement la diversité d'intérêts dans la liste de recommandations.

Alternativement, il est possible de sélectionner les meilleurs représentants des différentes mesures dans des proportions homogènes. C'est ce type d'agrégation que

nous privilégions en faisant l'hypothèse que, de cette façon, les différents aspects des intérêts des usagers seront couverts équitablement. Nous allons le montrer expérimentalement.

Notre proposition ne se base donc ni sur une combinaison des scores, qui ne sont pas nécessairement comparables et normalisés, ni sur l'utilisation des coefficients censés refléter l'importance des mesures. Nous avons en effet choisi d'intervenir sur la représentativité de chaque mesure de similarité dans la liste de recommandations. Plutôt que de nous appuyer sur la combinaison des scores obtenus par chaque document, nous avons fait le choix de sélectionner les meilleurs représentants des différentes mesures dans des proportions variables, et ainsi garantir une plus grande diversité des résultats.



Gâteau au chocolat

Voici une recette toute simple de gâteau au chocolat avec un bon goût de chocolat. Quand j'ai vu cette recette dans le magazine Marmiton, j'ai été accroché par l'utilisation de lait de coco. On ne sent pas du tout le lait de coco, celui ci renforce juste le parfum du chocolat. Ce gâteau est bien moelleux (pas étouffe chrétiens d'après mon mari!! lol)
et accompagné d'une crème anglaise maison c'est un plaisir. De la simplicité après les fêtes!!

Ingrédients pour un moule de 20×20 cm

- 200 ml de lait de coco
- 300 g de cassonade
- 100 g de farine
- 3 oeufs
- 100 g de beurre
- 200 g de chocolat pâtissier (Nestlé noir corsé)
- 1/2 sachet de levure

Réalisation:
Préchauffer le four à 180°C.
Dans une casserole faire fondre à feu doux, le chocolat, le beurre et le lait de coco.
Dans un saladier mélanger la farine, la levure, et la cassonade. Ajouter les oeufs un par un et ajouter le mélange chocolaté précédent en mélangeant entre chaque ajout.
Verser dans un moule à bords hauts et faire cuire environ 45 minutes (il était dit dans la recette 1h15 de cuisson mais cela me semble beaucoup trop si on souhaite un gâteau moelleux et pas sec), la lame d'un couteau doit ressortir sèche.
Démouler une fois refroidie et servir avec une crème anglaise maison (si possible).

FIGURE 9 – Document initial utilisé pour contruire la liste de recommandations

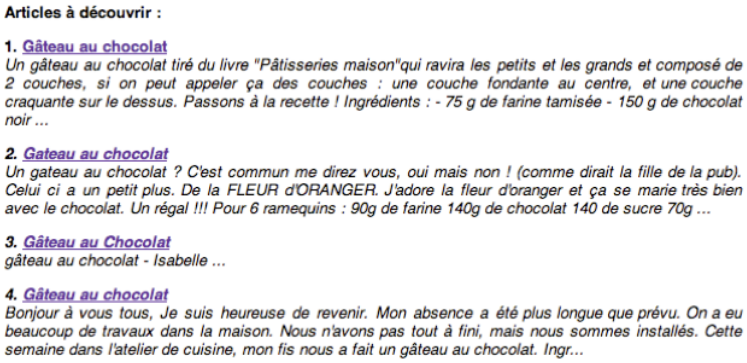


FIGURE 10 – Recommandations issues d'un système de recommandation n'utilisant qu'une seule mesure de sélection (par contenu)

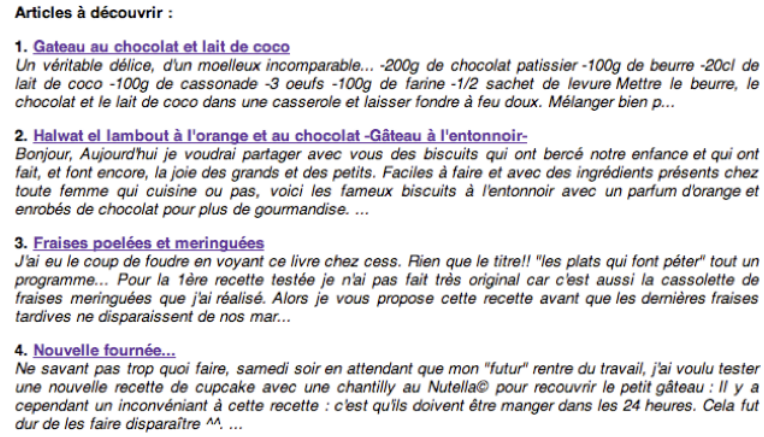


FIGURE 11 – Recommandations issues d'un système de recommandation reposant sur notre modèle et utilisant quatre mesures de sélection

Les figures 10 et 11 illustrent ce mécanisme en présentant les recommandations obtenues pour un article issu de la plateforme *OverBlog* et décrivant une recette de gâteau au chocolat (cf. figure 9).

La figure 10 présente les recommandations obtenues avec un système de recommandation se limitant à une mesure de similarité basée sur le contenu. Nous avons simulé ce système de recommandation à partir d'un outil de recherche d'information en utilisant le titre du document visité. La figure 11 montre quant

à elle les recommandations obtenues à l'aide d'un système de recommandation basé sur notre modèle et utilisant quatre mesures de similarité différentes. Chaque résultat correspond au meilleur représentant d'une des mesures de similarité présentées en section 4.4.2.2 :

- **similarité de contenu** : les recommandations sont obtenues par une recherche effectuée grâce à l'outil *Apache Solr* ;
- **similarité de contenu combinée à l'algorithme des k-moyennes** : l'objectif de la phase de classification est d'identifier les sous-aspects du thème du document visité ;
- **similarité organisationnelle** : les recommandations sont sélectionnées aléatoirement parmi les articles appartenant au même blog que le document visité ;
- **similarité organisationnelle combinée à une mesure liée à la popularité** : les recommandations sont extraites aléatoirement des articles les plus populaires de la catégorie du blog d'appartenance du document visité (catégorie "Cuisine" dans le cas de notre exemple).

Dans le cas du système de recommandation de la figure 10, les recommandations sont toutes très similaires et ne satisferont qu'un usager souhaitant d'autres recettes de gâteau au chocolat. Les recommandations obtenues avec notre modèle (figure 11) sont plus diversifiées puisqu'en plus d'une recette alternative de gâteau au chocolat (recommandation 1), l'usager obtient des recettes plus éloignées (recommandations 2, 3 et 4) de celle proposée sur le document visité. Cette seconde liste de recommandations permet de répondre aux intérêts d'un panel plus large d'usagers.

4.5.3 Ré-ordonnancement des résultats

Suite à la sélection de documents selon différentes mesures de sélection et à leur agrégation au sein d'une liste unique, l'ordonnancement des recommandations s'effectue selon deux tris successifs :

- tout d'abord, les documents les plus représentés, c'est-à-dire ceux présents dans les résultats de mesures différentes, sont positionnés en tête de classement. Il s'agit là de respecter l'effet "Chorus" qui stipule qu'il peut y avoir accord de différentes techniques sur la pertinence d'un document (Vogt et Cottrell, 1999) ;
- dans un second temps, les documents étant présents dans le même nombre de résultats de mesures de sélection sont réordonnés selon l'IAIR (*Inverse Average Inverse Rank* (Siegler *et al.*, 1997)) des documents, c'est-à-dire en fonction du rang qu'ils occupent dans les résultats de leur(s) mesure(s) d'origine.

L'IAIR est défini de la manière suivante :

$$IAIR = \frac{1}{\sum_i (rank_i^{-1})}$$

où $rank_i$ est le rang du document i.

En considérant le nombre de mesures d'origine ainsi que le rang des documents, nous nous affranchissons d'une phase de normalisation des scores qui peuvent, selon les mesures de sélection, être définis sur des ensembles de valeurs hétérogènes.

Selon notre approche, les documents présents dans plusieurs mesures sont donc favorisés lors du ré-ordonnancement des recommandations dans la mesure où ils répondent à plusieurs intérêts et qu'ils ont donc plus de chance de satisfaire un grand nombre d'usagers.

4.6 Processus d'apprentissage

A l'issue du processus d'agrégation, nous proposons à l'usager une liste de recommandations diversifiées. Afin de tendre vers les intérêts que le document satisfait réellement, il est nécessaire d'ajuster la proportion de documents extraits de chaque mesure. En effet, nous supposons qu'un document suscite ou répond à certains intérêts plus qu'à d'autres. Pour identifier ces intérêts émergents qui conduisent à une satisfaction accrue de l'usager, il convient de favoriser les mesures qui restituent les documents réellement jugés comme pertinents par les usagers. Pour ce faire, il est nécessaire de capter et d'analyser les jugements d'intérêt, qu'ils soient explicites (notation des résultats) ou implicites (clics).

L'ajustement des proportions de documents provenant de chaque mesure de sélection est alors fonction de l'analyse des jugements d'intérêt. Cette analyse est confiée à ce dernier composant.

Le composant *processus d'apprentissage* doit permettre l'amélioration du processus d'agrégation sans émettre d'hypothèse sur les intérêts réels de l'usager que nous ignorons. Nous nous intéressons plutôt à savoir si des intérêts particuliers se dégagent de la visite d'un document. Si tel est le cas, le système peut modifier la proportion de recommandations issues de chaque mesure de sélection.

Lorsqu'un usager choisit une recommandation, nous pouvons supposer qu'elle répond à l'un de ses intérêts. Le processus d'apprentissage identifie la ou les mesures de sélection à l'origine de cette recommandation. Pour un document donné, le processus d'agrégation est donc capable de déterminer quelles mesures ont conduit à la consultation d'une recommandation. Il est donc capable d'ajuster la représentativité de chaque mesure, c'est-à-dire la proportion de recommandations issues

de chacune d'entre elles dans la liste de recommandations finale. Lorsqu'un document est obtenu au travers de différentes mesures, les proportions sont ajustées pour l'ensemble de ces mesures.

Le choix d'une recommandation par un usager se caractérise par un clic. En accord avec la littérature (cf. section 1.4.3), nous avons choisi d'utiliser ce jugement de pertinence implicite pour traduire un lien entre le document visité et la recommandation. Afin de limiter autant que possible la phase d'analyse des clics, et de permettre ainsi le passage à l'échelle de notre modèle dans un contexte industriel, nous avons adopté un modèle relativement simple. Nous considérons en effet qu'un clic correspond à un jugement d'intérêt, et ce sans tenir compte de la position de la recommandation dans la liste finale.

La proportion $P(x_i)$ de recommandations dans la liste finale (agrégée) est déterminée par la formule suivante :

$$P(x_i) = \frac{\alpha + x_i}{\alpha N + X}$$

où N est le nombre de mesures de sélection utilisées, X le nombre total de clics et x_i le nombre de clics d'une mesure de sélection. α est défini ainsi :

$$\alpha = Max(0, C - X)$$

La constante C précise le nombre de clics nécessaire avant de pouvoir envisager l'éviction d'une mesure (par exemple $C = 100$). X et x_i peuvent être considérés à plusieurs niveaux :

— globalement, c'est-à-dire au niveau des mesures de sélection ;
— au niveau de l'usager, par la construction d'un profil définissant ses intérêts ;
— ou au niveau du document visité.

Nous privilégions cette dernière stratégie qui vise à identifier les intérêts qu'un document visité suscite sur la base des recommandations associées sélectionnées. Nous établissons en quelques sortes un profil de document.

Les mesures de sélection non adaptées, en d'autres termes pour lesquelles les recommandations associées ne sont jamais consultées, peuvent être éliminées (dans le cas où $x_i = 0$, si et seulement si $\alpha = 0$).

Le fait de sélectionner les meilleurs représentants de chaque mesure de sélection au moment de l'agrégation rend l'apprentissage plus discriminant et permet de tendre plus rapidement vers les intérêts supposés des visiteurs d'un document.

4.7 Expérience utilisateur : perception et intérêt de la diversité

Les protocoles d'évaluation existants, comme la tâche *diversité* de la campagne TREC Web, ont été conçus pour évaluer les performances des approches basées sur le contenu textuel des documents. Ils ne sont pas adaptés pour l'évaluation d'autres types de diversité ("sérendipité" par exemple) pour lesquels la satisfaction des usagers ne peut être réellement appréciée qu'au travers d'une expérience utilisateur (Hayes *et al.*, 2002).

L'évaluation envisagée pour notre modèle doit répondre à trois objectifs :

Objectif 1 Permettre l'analyse de la pertinence des recommandations et de la perception de la diversité par les usagers (étude qualitative).

Objectif 2 Démontrer que les recommandations proposées sont utilisées par les usagers (étude quantitative) et que notre modèle conduit à une utilisation accrue de ces recommandations.

Objectif 3 Permettre d'estimer l'incidence globale du système de recommandation lorsqu'il est intégré à une plateforme de contenus (étude d'impact).

Cette évaluation implique par conséquent un panel d'usagers, ainsi qu'une quantité importante de documents les plus diversifiés possibles. Les plateformes de blogs comme *OverBlog* constituent un terrain d'expérimentation idéal en permettant l'évaluation du système de recommandation au travers de jugements réels d'usagers comme le préconisent (Hayes *et al.*, 2002).

Cette section ne traite que de l'analyse qualitative des recommandations (objectif 1). Le chapitre 5 est quant à lui consacré aux objectifs 2 et 3.

Nous avons mené une expérience utilisateur pour nous assurer de la pertinence de notre modèle, qui propose des recommandations diversifiées tout en maintenant un bon niveau de précision.

Cette évaluation doit valider plusieurs hypothèses :

Hypothèse 1 La plupart du temps, les usagers souhaitent obtenir de l'information thématiquement liée aux contenus qu'ils consultent ("topicalité").

Hypothèse 2 Parfois, les usagers veulent étendre leur connaissance d'un sujet qui les intéresse (diversité de contenu).

Hypothèse 3 Quelques usagers sont dans un processus de découverte et recherchent de l'information nouvelle, non nécessairement liée au contenu visité ("sérendipité").

Pour vérifier ces hypothèses, nous avons mis en place une plateforme d'évaluation reposant sur notre modèle et nous l'avons soumise à un groupe d'usagers.

4.7.1 Protocole expérimental

Le panel d'usagers se compose de trente quatre étudiants en Master de Management, parlant tous couramment français. Nous leur demandons de tester et de comparer différents systèmes de recommandation. La durée de l'évaluation est fixée à une heure et comporte six tâches.

Chaque tâche se décompose comme suit :

— L'usager saisit une requête en langage naturel dans un moteur de recherche et obtient une liste de résultats dans laquelle il doit choisir un document qui lui semble intéressant. Ce document sera à l'origine des recommandations.
— Une fois le document choisi, l'usager le consulte dans son intégralité.
— A l'issue de la lecture du document, deux listes comportant cinq recommandations chacune lui sont proposées. Une de ces listes est choisie aléatoirement parmi l'une des mesures de sélection de notre système et reflète des intérêts particuliers. Ces mesures, considérées individuellement, constituent nos bases de référence. La seconde liste de recommandation provient quant à elle de notre modèle : elle agrège les résultats des cinq mesures du système.
— L'usager indique la liste qui lui parait être la plus pertinente.
— Il précise ensuite celle qui lui semble la plus diversifiée.
— Enfin, les deux listes sont fusionnées et l'usager doit indiquer les documents qu'il juge pertinents.

Pour la première et la quatrième tâches d'évaluation, la requête et le document sont imposés. Pour la seconde et la cinquième tâche, la requête est imposée et le document est librement choisi. Pour la troisième et la sixième tâche, la requête et le document sont libres.

La sous-section suivante présente le corpus, ainsi que l'architecture de la plateforme d'évaluation.

4.7.2 Plateforme d'évaluation : architecture et corpus utilisé

Pour cette expérimentation, nous nous sommes focalisés sur des documents rédigés en français et issus de la plateforme de blogs *OverBlog*. Ces données représentent plus de vingt millions de billets provenant de près d'un million de blogs distincts.

Nous avons utilisé les cinq mesures de sélection présentées en section 4.4.2.2 sur ce corpus pour obtenir plusieurs listes de recommandations à agréger.

La figure 12 présente l'architecture de la plateforme d'évaluation. A partir de

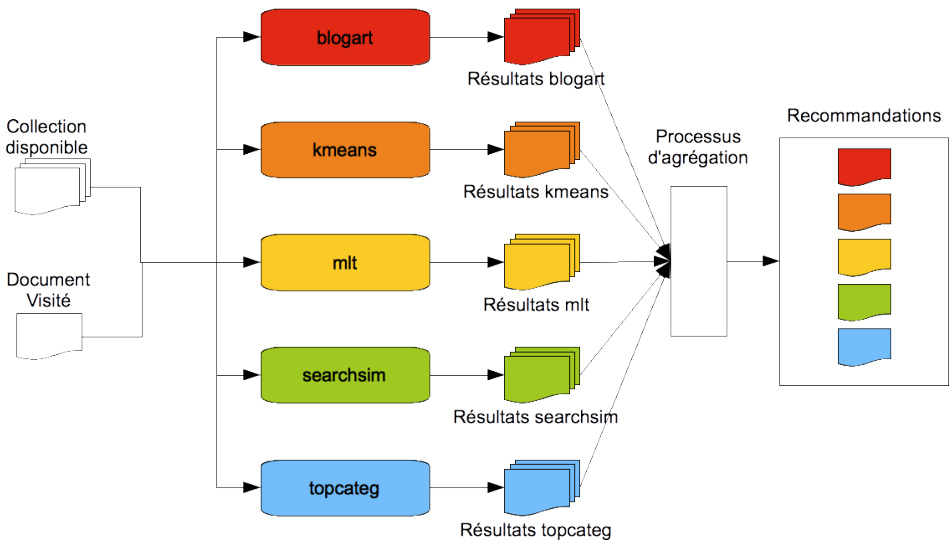


FIGURE 12 – Architecture de la plateforme d'évaluation

la collection disponible et du document visité, chaque mesure restitue indépendamment un ensemble ordonné de documents. Ces résultats sont les données en entrée du processus d'agrégation qui sélectionne le meilleur document de chaque ensemble. La liste finale comporte par conséquent cinq documents, un par mesure de similarité. Ce nombre restreint de recommandations (cinq) proposées à l'usager a été conditionné par la durée de l'évaluation fixée à une heure. Un nombre plus important de recommandations à évaluer nous aurait contraints à réduire le nombre de requêtes.

L'utilisation des cinq mesures de sélection simulent les différents types de diversité et vise à limiter le chevauchement des documents qu'elles restituent. Afin de vérifier que les mesures conduisent effectivement à des ensembles de documents différents, nous avons calculé le chevauchement de la même manière que lors de l'étude préliminaire présentée en section 4.2. Nous observons sur la figure 13 la même tendance que lors des expérimentations menées sur les tâches *adhoc* et *diversité* :

— le chevauchement est faible pour les mesures basées sur le contenu (*mlt*, *searchsim* et *kmeans*) ;
— il est nul pour les autres mesures (*blogart*, *topcateg*).

Le processus d'apprentissage n'a quant à lui pas été évalué au cours de cette expérience utilisateur.

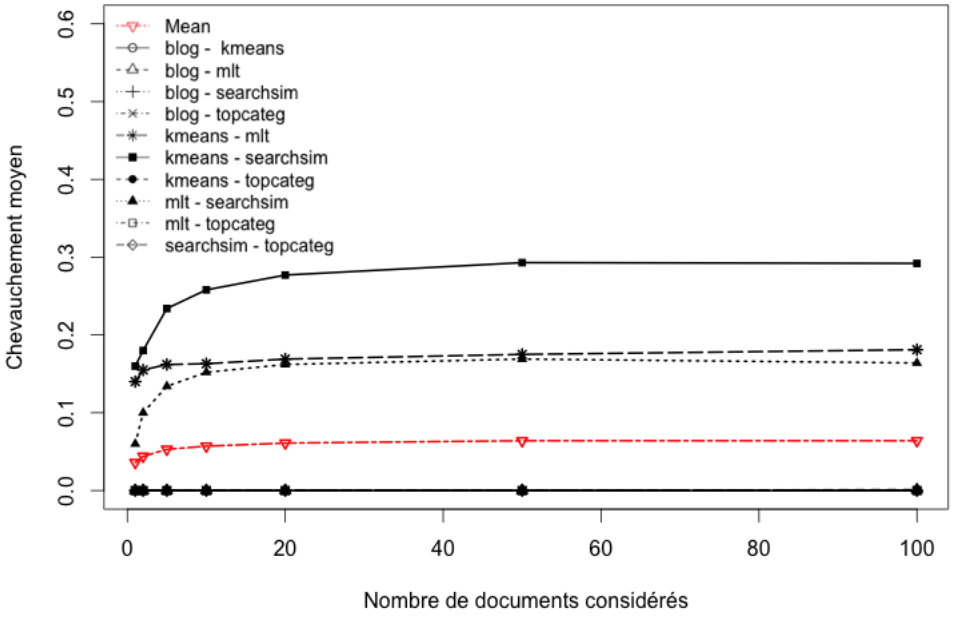


FIGURE 13 – Chevauchement moyen entre les listes de recommandations obtenues par les mesures d'intérêts utilisées sur le corpus *OverBlog*

4.7.3 Résultats

Le tableau 4 synthétise le retour des usagers concernant la pertinence des listes de recommandations, ainsi que leur perception de la diversité de ces listes. Par exemple ($4^{ème}$ ligne), 76.5% des listes obtenues avec la mesure *mlt* ont été considérées comme plus pertinentes que la liste agrégée. Nous notons que les mesures perçues comme les plus pertinentes sont celles qui se focalisent sur les liens thématiques.

En moyenne, les recommandations issues de notre modèle sont décrites comme plus pertinentes que les autres mesures dans près d'un cas sur deux. Nous obtenons un résultat similaire pour la mesure *blogart*. Ceci est plus surprenant, mais confirme que les attentes des usagers ne se focalisent pas nécessairement sur le seul lien thématique entre documents.

Les résultats obtenus pour la question "Quelle liste vous semble la plus diversifiée ?" sont plus surprenant : il n'y a pas de différences marquantes entre les mesures, et la mesure agrégée est vue par les usagers comme plus diversifiée dans

50% des cas. Nous pensons que cela s'explique par le fait que les usagers éprouvent des difficultés à définir la notion de diversité. Nous aurions probablement pu les aider en clarifiant la question posée.

Mesure de sélection	**Pertinence**	**Diversité**
blogart	44,7%	55,3%
kmeans	70,8%	33,3%
mlt	76,5%	50,0%
searchsim	64,3%	42,9%
topcateg	15,4%	65,4%
Moyenne	54,34%	49,4%

TABLE 4 – Pourcentage des usagers qui considère une mesure de sélection particulière comme plus pertinente/diversifiée que la mesure agrégée

Le tableau 5 montre la précision de chaque mesure de sélection, c'est-à-dire la proportion de documents pertinents dans la liste de recommandations. Ces résultats confirment que les approches basées sur le contenu sont perçues comme plus pertinente. Par exemple la mesure *kmeans*, qui traduit la diversité de contenu, obtient les meilleurs résultats. Au contraire, les mesures *topcateg* et *blogart* qui visent la "sérendipité" conduisent à de moins bons résultats.

Mesure de sélection	**Précision**
blogart	0.147
kmeans	0.385
mlt	0.265
searchsim	0.307
topcateg	0.038
agrégée	0.267

TABLE 5 – Précision par mesure de sélection

La liste de recommandations agrégée offre un compromis intéressant aux différentes mesures et présente un bon équilibre entre diversité et précision. En effet, la valeur de la précision est de 0,267 ce qui est supérieur à la précision moyenne des autres mesures (0,228).

Même si ce score est inférieur à la meilleure des mesures (*kmeans*), les résultats sont encourageants, d'autant plus que la précision des mesures *topcateg* et *blogart*

est très faible. En effet, ces deux mesures peuvent introduire du bruit dans les recommandations, ce qui par conséquent pénalise la précision globale. Cependant, cette perte de précision était prédictible puisque les recommandations agrégées sont plus diversifiées (dans 50% des cas en moyenne). Ces effets négatifs de la diversité sur la précision ont déjà été soulignés par Adomavicius et Kwon (2012).

Finalement, le tableau 6 permet de comparer la mesure agrégée aux autres mesures. Il donne la proportion de documents pertinents restitués par chacune des mesures comparées. Par exemple, lorsque la mesure *mlt* a été comparée à la mesure agrégée ($4^{ème}$ colonne), 54,69% des documents pertinents n'étaient restitués que par la mesure *mlt*, 32,81% par la mesure agrégée uniquement, et 12,50% par les deux mesures. Nous pouvons observer que, même si la majorité des documents pertinents provient des mesures basées sur le contenu, une part significative de ces documents est issue de la mesure agrégée. En effet, plus de 20% des documents pertinents ne sont restitués que par notre modèle, et ils n'auraient pas été proposés à l'usager en utilisant une seule mesure. Ceci justifie donc pleinement notre approche.

4.7.4 Conclusions de l'expérience utilisateur

La mesure agrégée reposant sur notre modèle offre un cadre permettant l'agrégation de multiples mesures de sélection afin de produire une liste de recommandations diversifiée. Bien que le modèle implanté ne surpasse pas les autres mesures en termes de précision, il permet de favoriser la diversité. De plus, en diversifiant les recommandations, notre proposition permet de répondre à un éventail plus large d'intérêts, tandis que les autres approches se focalisent sur la majorité des besoins des usagers, à savoir la similarité de contenus.

Les mesures utilisées pour traduire la sérendipité sont relativement simples. Cependant, les résultats qu'elles produisent ont été considérés comme pertinents par les usagers. Ceci est particulièrement intéressant et traduit le fait que les besoins des usagers peuvent être mieux satisfaits par différents types de diversité.

Le modèle ayant été évalué de manière qualitative, il convient à présent de nous assurer de sa pertinence dans un contexte industriel à grande échelle. Pour cela, nous avons implanté notre modèle au sein de la plateforme *OverBlog*. Cette seconde expérimentation doit également nous permettre d'évaluer l'impact d'un système de recommandation pour ce type de plateforme de contenus.

Mesures : *aggregée* comparée à	***blogart***	***kmeans***	***mlt***	***searchsim***	***topcateg***
Documents restitués uniquement par la mesure seule	35,00%	52,46%	54,69%	52,43%	8,77%
Documents restitués uniquement par la mesure agrégée	65,00%	21,31%	32,81%	38,83%	91,23%
Documents communs	0,00%	26,23%	12,50%	8,74%	0,00%

TABLE 6 – Distribution des documents pertinents

Chapitre 5

L'implantation du modèle au sein de la plateforme *OverBlog*

5.1 Introduction

Les plateformes de blogs tirent la majorité de leurs revenus de la publicité. Ces revenus sont directement liés à l'audience. Le taux de clics publicitaires se situant généralement entre 0,1% et 1% [1], il est nécessaire de générer un trafic très important pour maintenir un niveau de revenu suffisant.

Les systèmes de recommandation sont donc un outil à considérer pour accroître le trafic de manière significative. L'objectif est de capter l'usager et de le maintenir captif sur la plateforme en lui proposant des recommandations capables de satisfaire ses intérêts. Il est de plus important de s'affranchir, autant que faire se peut, des sources de trafic extérieures (moteurs de recherche commerciaux, réseaux sociaux, ...) afin de ne pas mettre en péril la pérennité de l'entreprise en cas de défaillance d'une de ces sources.

Dans le contexte des plateformes de blogs, la grande majorité des visiteurs sont des utilisateurs étrangers à la plateforme. Sur la plateforme *OverBlog* qui recense près de 80 millions de visiteurs uniques par mois, moins de 2% d'entre eux sont des utilisateurs de la plateforme, c'est-à-dire possédant un compte et disposant d'un profil de blogueur. Ce déficit d'information sur les usagers rend les approches de type filtrage collaboratif inutilisables.

Parallèlement à cette problématique liée aux usagers, les plateformes de blogs sont confrontées à l'hétérogénéité des billets publiés. Il est ainsi très difficile

1. Étude Mediamind sur les performances des campagnes publicitaires selon les formats d'affichage :
www2.mediamind.com/data/uploads/resourcelibrary/MediaMind_Benchmark_H1_2012.pdf

d'identifier des caractéristiques communes aux différents billets, tant leurs contenus et leurs structures diffèrent. Les approches de type filtrage basé sur le contenu sont donc elles aussi inappropriées. La masse d'information à traiter (environ 20 millions de billets sur *OverBlog* au moment de nos travaux) rend la tâche d'autant plus complexe.

C'est dans cet environnement que notre modèle prend tout son sens puisqu'il ne nécessite aucune connaissance préalable de l'usager. La diversité des mesures de sélection agrégées permet quant à elle de faire face à la disparité des billets et des préférences utilisateur.

L'objectif de ce chapitre est de démontrer la viabilité de notre proposition dans un contexte industriel à grande échelle. La viabilité de l'approche est évaluée tant au niveau des performances en termes de qualité des recommandations, que d'un point de vue opérationnel, en considérant notamment les temps de calcul et l'infrastructure nécessaires pour la mise en place du modèle.

Dans la section 5.2, nous évoquons les contraintes industrielles qui nous ont été imposées lors de l'intégration de notre modèle au produit *OverBlog*. Nous présentons ensuite dans la section 5.3 les outils de supervision mis en place afin de contrôler le déroulement des expérimentations et de nous assurer qu'elles ne mettaient pas en péril le produit existant. Ces outils ont également pour objectif d'analyser quotidiennement les performances des recommandations. Dans la section 5.4, nous nous attardons sur la préparation des données nécessaires aux mesures de sélection et permettant de garantir la qualité des recommandations proposées aux usagers. Nous nous focalisons ensuite dans la section 5.5 sur l'architecture de la plateforme. Nous mettons en évidence l'intégration de notre proposition au sein de l'architecture existante. Enfin, nous présentons les résultats obtenus dans la section 5.6.

5.2 Contraintes industrielles

Le système de recommandation étant intégré à un produit en exploitation, nous avons été soumis à un certain nombre de contraintes visant à garantir la qualité de service de la plateforme de blogs.

Nous avons dû dans un premier temps intégrer notre modèle sous la forme d'un service interrogé à la demande par le produit au travers d'un appel JavaScript [1]. Ceci s'explique par le fait que les pages Web générées à l'aide du langage PHP [2] ne sont pas affichées avant qu'elles ne soient entièrement construites, alors qu'un

1. https ://developer.mozilla.org/fr/docs/JavaScript
2. http ://www.php.net

appel JavaScript intervient de manière asynchrone. Par conséquent, il n'est pas bloquant pour l'affichage des pages. En cas de défaillance du service, ou si le temps de réponse dépasse un certain seuil (expiration de l'appel), les recommandations ne seront pas affichées mais cela n'aura pas d'impact sur le reste de l'affichage des blogs.

Afin d'éviter ces problèmes d'expiration, la seconde contrainte imposée est un temps de réponse de notre service inférieur à la seconde. Cela permet également d'éviter de consommer trop longtemps des ressources (temps de calcul, mémoire) qui pourraient être nécessaires à un autre processus. Cette contrainte nous a conduit à l'abandon des mesures de sélection *kmeans* et *topcateg* présentées dans notre proposition théorique en section 4.4.

La dernière contrainte concerne la maîtrise de la charge induite par l'activation du service de recommandation. Les serveurs supportent mal les montées en charge brutales lors de l'activation d'un nouveau service. Il est par conséquent vital d'assurer une montée en charge progressive pour laisser aux serveurs le temps de s'optimiser. Pour répondre à cette contrainte, nous avons mis en place deux paramètres de configuration gérant l'activation des recommandations :

— un premier paramètre permet de définir un groupe de blogs cibles pour l'affichage des recommandations. Trois groupes sont définis pour la plateforme *OverBlog* : les blogs "délaissés", c'est-à-dire sans activité depuis plus de 45 jours, les blogs "Premium" pour lesquels les clients payent un ensemble de services additionnels, et les blogs qui n'entrent dans aucune des deux catégories précédentes. Un quatrième groupe englobant les trois précédents est également défini.

— le second paramètre permet de se restreindre à un sous-ensemble du groupe précédemment choisi. Pour cela, le modulo de l'identifiant du blog est utilisé. Seul les blogs dont le modulo est égal à 1 sont utilisés pour l'expérimentation. Ainsi, en fixant le modulo à 10, 10% des blogs seront concernés. En le positionnant à la valeur 2, 50% des blogs proposeront des recommandations selon notre approche.

Ce principe est particulièrement intéressant également dans la phase d'évaluation pour mesurer l'impact de nos propositions sur une partie du système pendant que l'autre partie fonctionne sans nos propositions. La configuration des recommandations peut être mise à jour à n'importe quel moment sans qu'aucune modification du code source ne soit nécessaire, et ce afin de pouvoir intervenir en temps réel sur le service.

5.3 Outils de supervision

Afin de suivre le déroulement des expérimentations, des outils de supervision ont été mis en place sur la plateforme *OverBlog*. Ces outils peuvent être classés en deux groupes :

— ceux permettant le contrôle de la charge opérationnelle ;

— et ceux assurant le suivi des performances des recommandations.

Ces outils sont développés à l'aide de technologies Web et sont présentés sous forme de graphiques pour en faciliter la lecture. Les équipes techniques et de recherche peuvent donc y accéder depuis un simple navigateur Web.

Les outils de contrôle de la charge opérationnelle étant déjà en place à mon arrivée, je n'ai assuré que le développement des outils spécifiques aux recommandations.

Cette section se contente de présenter les graphiques utilisés lors de l'expérimentation. Ils seront analysés dans la section 5.6, consacrée aux résultats.

5.3.1 Contrôler la charge opérationnelle

Cette première série d'outils a pour objectif d'évaluer l'impact du modèle sur l'infrastructure de la plateforme *OverBlog*. Il ne s'agit pas d'outils spécifiquement mis en place pour nos expérimentations, mais d'indicateurs développés et utilisés au quotidien par les équipes techniques d'*OverBlog* pour surveiller la plateforme dans son ensemble.

Ces graphiques, développés grâce au langage PHP et à l'outil RRD [1], sont mis à jour toutes les cinq minutes et présentent les données pour la journée, la semaine, le mois et l'année courants. Les données de chaque serveur supervisé sont collectées par l'intermédiaire d'un script Perl [2] exécuté sur ces serveurs et sont centralisées sur un serveur de supervision.

5.3.1.1 Charge des serveurs de bases de données

Le graphique présenté en figure 14 indique la charge CPU des différents serveurs de bases de données. La charge CPU correspond au nombre de processus mis en attente faute de ressources disponibles. Elle est jugée acceptable si elle est inférieure au nombre de cœurs disponibles sur le serveur.

La courbe rouge indique la charge actuelle. La courbe violette indique la charge de la semaine passée et fait office de référence pour faciliter les comparaisons en

1. http ://oss.oetiker.ch/rrdtool/
2. http ://www.perl.org/

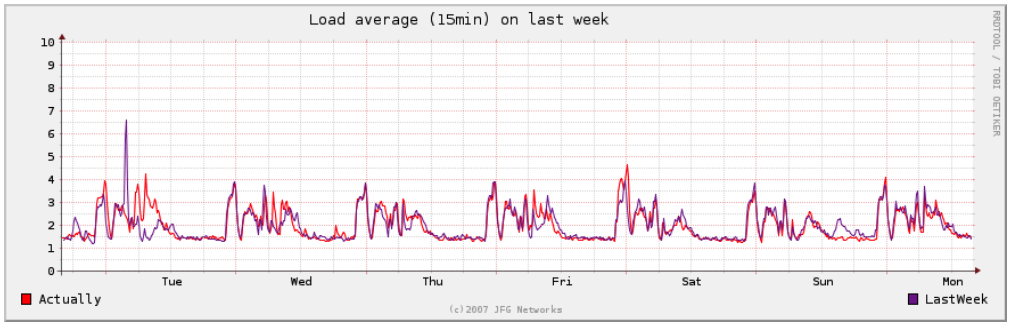


FIGURE 14 – Suivi de la charge hebdomadaire d'un serveur de bases de données

cas d'écart à la normale.

Il est également possible de visualiser la charge cumulée des serveurs afin d'avoir une vision globale de l'environnement de production (figure 15).

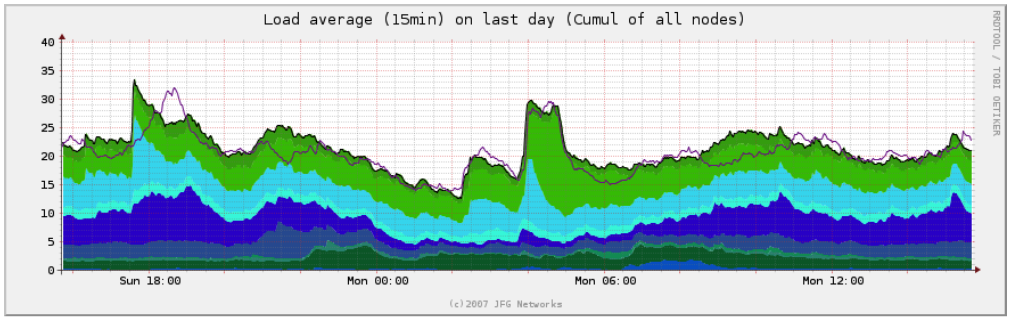


FIGURE 15 – Suivi de la charge cumulée quotidienne des serveurs de bases de données

5.3.1.2 Charge des serveurs frontaux

Similairement aux graphiques de la charge des serveurs de bases de données, le graphique en figure 16 présente la charge d'un serveur frontal. Il s'agit du serveur en charge de la construction des pages Web et de leur transmission au client qui les affichera. Il est également possible de visualiser la charge cumulée.

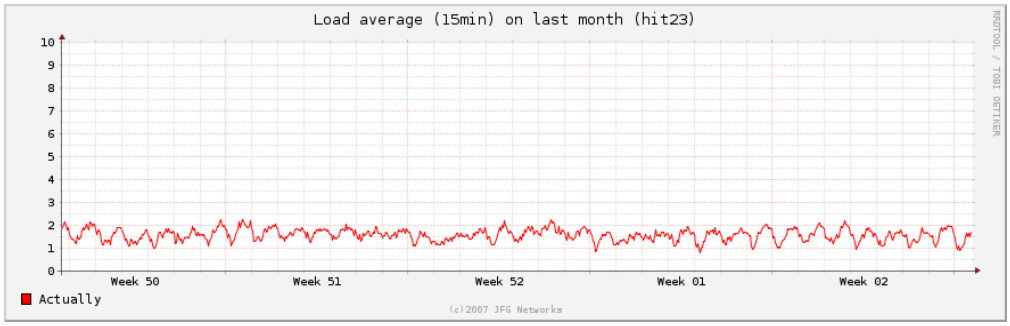


FIGURE 16 – Suivi de la charge mensuelle d'un serveur frontal

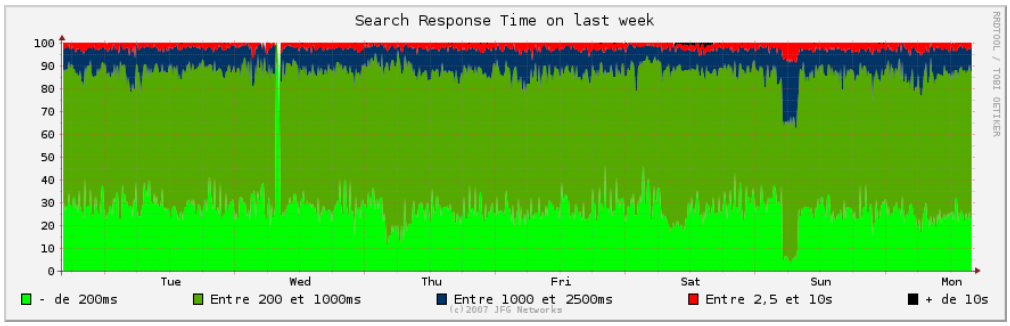


FIGURE 17 – Temps de réponse moyen du moteur de recherche sur une période d'une semaine

5.3.1.3 Temps de réponse des serveurs de recherche d'information

Le graphique présenté en figure 17 indique la proportion de requêtes soumises au moteur de recherche qui aboutissent dans un certain intervalle de temps. Sur cet exemple, nous pouvons voir qu'en moyenne les requêtes sont traitées :

— pour 25% en moins de 200ms ;
— pour 85% en moins de 1000ms ;
— pour plus de 95% en moins de 2500ms ;
— et pour moins de 5% en plus de 2500ms.

Une rupture importante au niveau du graphique est caractéristique d'une défaillance du service et implique généralement une opération de maintenance.

5.3.2 Suivre les performances des recommandations

En complément de ces outils de supervision dédiés à l'infrastructure, j'ai mis en place des indicateurs spécifiques aux expérimentations afin d'évaluer les performances en termes de qualité de recommandation de notre modèle. Pour cela, j'ai utilisé les langages JavaScript et PHP ainsi que le service Google Charts [1] pour la génération des graphiques.

A chaque affichage de recommandations et à chaque clic, le serveur de supervision est notifié et stocke les informations nécessaires à l'analyse.

L'unité de temps utilisée pour les différents graphiques est la journée.

Les recommandations sont proposées à l'usager grâce à un pavé additionnel (bloc de recommandations) intégré à la suite du billet. Chaque bloc comporte six recommandations présentées sous la forme de vignettes associées au titre des billets recommandés.

5.3.2.1 Taux de clics global sur les recommandations

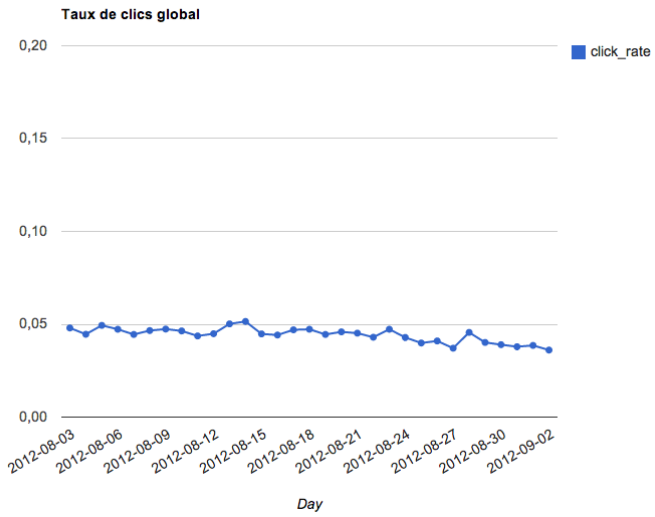


FIGURE 18 – Taux de clics global

Le premier indicateur disponible est le taux de clics global (cf. figure 18), toute mesure confondue.

1. https ://developers.google.com/chart/

Le taux de clics correspond au ratio du nombre de clics par le nombre d'affichages du bloc de recommandations. Ainsi, si chaque bloc de recommandations aboutit à un clic, le taux sera égal à 1.

5.3.2.2 Taux de clics par mesure de sélection

Le taux de clics est également calculé pour chacune des mesures de sélection employées lors des expérimentations. Afin de faciliter la comparaison entre les mesures, toutes les données sont affichées sur un même graphique, comme présenté en figure 19.

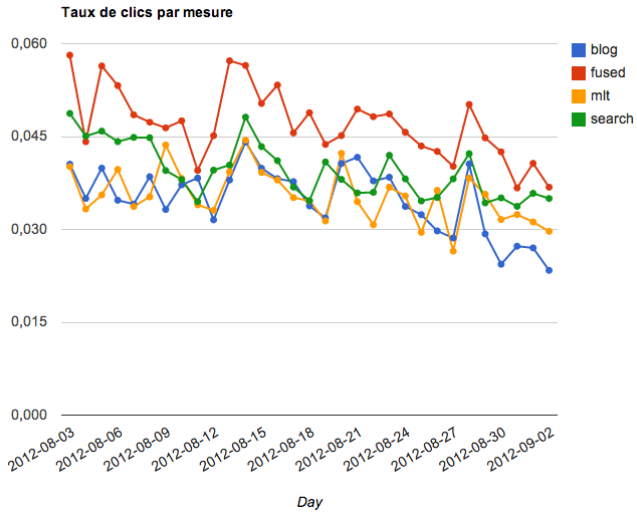


FIGURE 19 – Taux de clics par mesure de sélection

5.3.2.3 Position des clics

Le graphique présenté en figure 20 nous permet de suivre le positionnement des clics au sein du bloc de recommandations. Ce type d'indicateur est très utile pour comparer différentes solutions de présentation des recommandations, par exemple sous forme de listes ou de vignettes dans notre cas (figure 29).

5.3.2.4 Statistiques globales d'affichage

Les statistiques globales d'affichage (figure 21) indiquent le nombre global d'affichages, le nombre de blogs uniques d'où sont issues les recommandations,

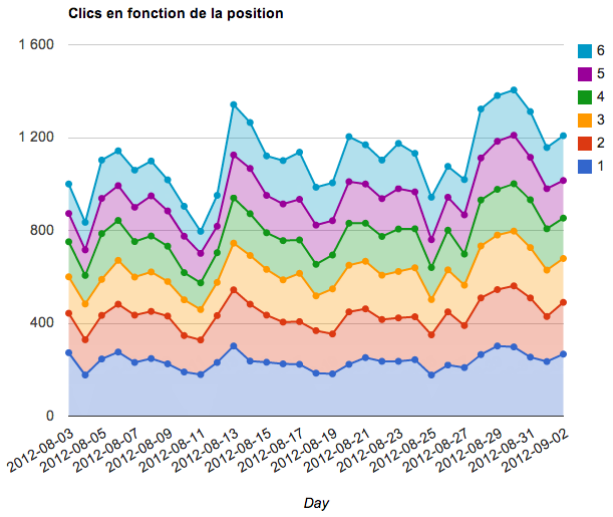


FIGURE 20 – Position des clics

le nombre d'articles uniques sur lesquels des recommandations ont été proposées aux visiteurs, ainsi que le nombre de visiteurs uniques.

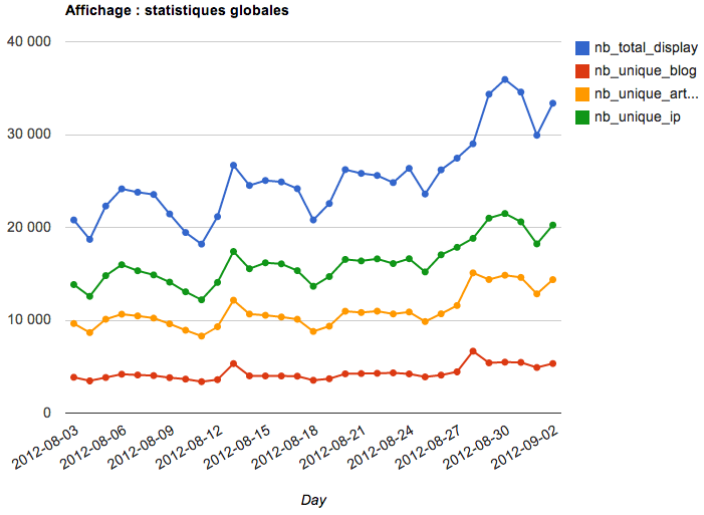


FIGURE 21 – Statistiques globales d'affichage

5.3.2.5 Nombre de blocs de recommandation calculés

En complément des indicateurs de charge, nous nous sommes intéressés au nombre de blocs de recommandation calculés quotidiennement (figure 22). Ce graphique est une base intéressante pour estimer les ressources requises par un nombre d'affichages plus important (modulé par les paramètres de configuration présentés en section 5.2) et facilite le dimensionnement de l'infrastructure matérielle.

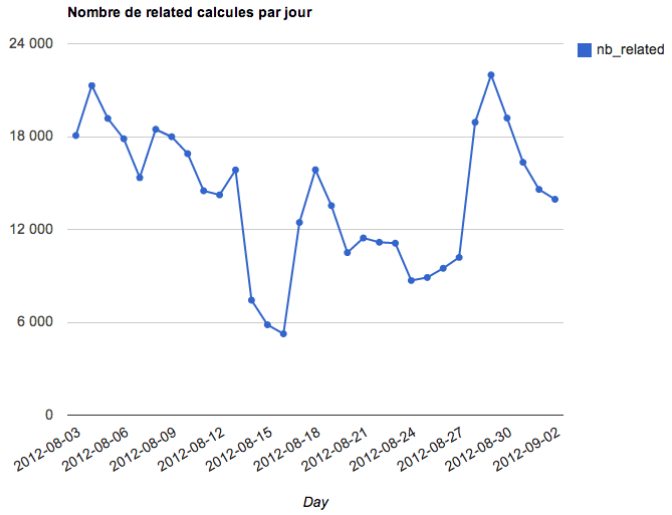


FIGURE 22 – Nombre de blocs de recommandation calculés quotidiennement

Ces indicateurs étant en place, une phase de préparation des données utilisées par les mesures d'intérêts a été nécessaire pour garantir un niveau de qualité suffisant des recommandations.

5.4 La préparation des données et des mesures de sélection utilisées

La principale source de données utilisée par les différentes mesures de sélection est l'index du moteur de recherche de la plateforme. Il permet en effet d'accéder à l'ensemble des caractéristiques des billets de blogs et d'évaluer la ressemblance en termes de contenu entre une requête et un billet, ou entre deux billets.

Le moteur de recherche de la plateforme tel qu'il était à mon arrivée dans l'entreprise souffrait de plusieurs problèmes :

— les temps de réponse étaient trop longs pour que les mesures de sélection que nous proposions soient en accord avec les contraintes industrielles imposées ;
— la qualité des résultats semblait inférieure à celle des approches de la littérature. Cette intuition a été confirmée par une évaluation ;
— enfin l'index était pollué par des contenus peu qualitatifs (*splogs* par exemple (Kolari *et al.*, 2006)) dont la valeur ajoutée pour l'usager est faible.

Il a donc été indispensable de corriger ces trois problèmes avant d'envisager d'implanter notre modèle.

Les solutions mises en œuvre sont présentées dans cette section. Nous nous focalisons dans un premier temps sur l'amélioration du moteur de recherche (section 5.4.1) qui doit aboutir à des meilleures performances au niveau des temps de réponse, pour garantir la viabilité des mesures de sélection d'un point de vue opérationnel, ainsi qu'au niveau de la qualité des résultats. Pour nous guider dans cette démarche d'amélioration, nous avons tout d'abord analysé et évalué le moteur de recherche existant. Les résultats obtenus ont ensuite été confrontés à la littérature pour nous conduire à un nouveau moteur de recherche reposant sur une solution open source.

Dans un second temps (section 5.4.2), nous nous intéressons brièvement à la lutte contre les *splogs* et à l'identification des contenus de mauvaise qualité afin d'obtenir une source de contenus qualitative.

5.4.1 Amélioration du moteur de recherche *OverBlog*

Afin de nous guider dans le processus d'amélioration du moteur de recherche, nous avons tout d'abord établi un état des lieux de l'existant, nécessaire pour en comprendre le fonctionnement et les motivations. Ce bilan a également pour objectif de mettre en exergue les différences avec les autres approches de la littérature afin d'identifier des axes d'amélioration.

5.4.1.1 Analyse de l'existant

Le moteur de la recherche interne à la plateforme *OverBlog* reposait initialement sur le module d'extension TSearch 2 [1] du SGBD (Système de Gestion de Base de Données) PostgreSQL [2]. Cette extension fournit au SGBD un processus d'indexation et des mesures de similarité applicables directement aux contenus textuels stockés en base de données.

Les sous-sections suivantes se focalisent sur le processus d'indexation et le

1. http ://www.sai.msu.su/ megera/postgres/gist/tsearch/V2/
2. http ://www.postgresql.org

modèle de recherche du moteur de recherche existant à mon arrivée.

Le processus d'indexation TSearch suit un processus d'indexation semblable à l'approche présentée dans la section 1.4. Le titre et le corps du billet sont les deux champs indexés. Le traitement est identique pour les deux champs :

- les termes sont tout d'abord extraits du contenu ;
- ensuite intervient l'élimination des "mots vides" ;
- les termes restants sont alors racinisés en utilisant l'algorithme de Porter (1980) ou ses variantes en fonction de la langue ;
- enfin, une phase d'analyse statistique associe chaque terme à l'ensemble de ses positions dans le document.

A l'issue de ce traitement, chaque document est représenté par deux vecteurs de termes : un pour le titre et un pour le corps. Les positions des termes sont utilisés par le modèle de recherche présenté ci-après.

Le modèle de recherche TSearch repose sur le modèle de recherche *Cover Density Ranking* proposé par Clarke *et al.* (2000) et présentant des performances proches du modèle BM25 (Robertson *et al.*, 1995). Le choix de ce modèle a été motivé par le fait que, contrairement à BM25 ou TF-IDF (Spärck Jones, 1972), il n'est pas nécessaire d'avoir accès à l'ensemble de la description du langage d'indexation (index inversé) pour calculer la similarité *(document, requête)*. L'implantation du modèle au sein du SGBD est alors facilitée. L'espace disque lié à la création des index inversés est également économisé.

L'ordonnancement des résultats s'effectue en deux temps. Tout d'abord, les documents sont regroupés en classes en fonction du nombre de termes distincts de la requête qu'ils possèdent. Les documents les plus pertinents seront ceux qui ont le plus de termes en commun avec la requête.

Les documents sont ensuite ordonnés pour chaque classe en déterminant leur densité de couverture. Pour chaque document, on détermine les sections (les couvertures) du document qui contiennent l'ensemble des termes en commun avec la requête. Les couvertures sont définies par la position du premier terme p_i et par la position du dernier terme q_i qui la composent. Une couverture (p_i, q_i) ne peut pas être contenue dans une autre.

Soit $\mathcal{C} = \{(p_1, q_1), (p_2, q_2), \ldots, (p_n, q_n)\}$ l'ensemble des couvertures d'un document. Le score du document est défini par :

$$S(\mathcal{C}) = \sum_{j=1}^{n} I(p_j, q_j)$$

Tel que :

$$I(p,q) = \begin{cases} \frac{\lambda}{q-p+1} & \text{si } q-p+1 > \lambda \\ 1 & \text{sinon} \end{cases}$$

Avec :

— (p_j, q_j) une couverture du document ;
— p_j la position d'un terme, q_j la position d'un autre terme telles que $p_j < q_j$;
— et λ une constante, dont la valeur est fixée à 4 dans (Clarke *et al.*, 2000).

Le score associé à un ensemble de couvertures repose sur deux hypothèses :

— plus la couverture est petite, plus il est probable que le texte associé à cette couverture soit pertinent ;
— plus le document possède de couvertures, plus la probabilité qu'il soit pertinent est importante.

En raison de la volumétrie considérable (environ 20 millions de billets lors de l'évaluation), l'index est découpé par trimestre, selon la date de création des billets. Une heuristique exploitant ce découpage a été mise en place pour tenter d'améliorer les temps de réponse. Ainsi on recherche d'abord les documents les plus récents, puis, si leur nombre n'est pas suffisant, la recherche se poursuit au sein des trimestres précédents.

5.4.1.2 Protocole d'évaluation de TSearch

Afin d'évaluer les performances du modèle de recherche TSearch, nous avons utilisé le corpus de référence TREC 8, ainsi que les outils TrecEval et Terrier, largement utilisés par la communauté de recherche d'information.

Le corpus TREC 8 Pour mener l'évaluation du moteur de recherche existant, nous avons utilisé le corpus de la campagne TREC 8 *adhoc*, constitué de 528155 documents en langue anglaise. L'évaluation porte sur 50 requêtes fournies avec le corpus et les jugements de pertinence (qrels). Plus de détails sont disponibles sur le site de TREC [1] .

TrecEval TrecEval [2] est un outil maintenu et utilisé par la communauté TREC pour évaluer les systèmes de recherche d'information. TrecEval permet de calculer différentes métriques couramment employées en recherche d'information (Précision, Rappel, MAP, ...) à partir de la liste des jugements de pertinence des documents par requête et de la liste des documents restitués par un système pour chacune de ces requêtes. L'outil permet en réalité de calculer la valeur de plus de

1. http ://trec.nist.gov
2. http ://trec.nist.gov/trec_eval/

100 mesures de performance sur chaque requête évaluée ; nous avons présenté dans le chapitre 3 les mesures que nous avons utilisées tout au long de ce travail.

La plateforme Terrier Terrier [1] est une plateforme Java open source développée et maintenue par l'Université de Glasgow. En plus d'un moteur d'indexation, Terrier implante plusieurs modèles de recherche (par exemple les modèles TF-IDF, BM25, ...) et permet donc de comparer les performances de ces modèles pour un même corpus et un même processus d'indexation.

5.4.1.3 Résultats de l'évaluation de TSearch

Suite à l'indexation du corpus, réalisée de manière analogue avec TSearch et la plateforme Terrier, nous avons calculé les métriques permettant d'évaluer les performances des différentes approches et de les comparer entre elles.

La figure 23 présente les courbes Rappel/Précision pour les trois modèles comparés. Nous constatons que les performances de TSearch sont en deçà de celles des modèles BM25 et TF-IDF. Ces deux derniers modèles ont quant à eux des performances très similaires (les courbes sont superposés).

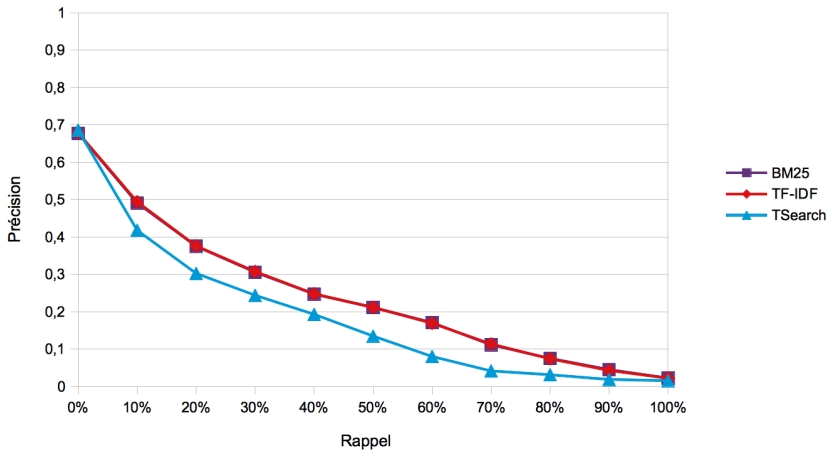


FIGURE 23 – Courbes Rappel/Précision des modèles TSearch, TF-IDF et BM25 pour le corpus TREC 8

Ces constatations se confirment au travers de la MAP (cf. figure 24) où nous notons un écart important entre les modèles de la littérature et celui utilisé par

1. http ://www.terrier.org

TSearch. Les résultats obtenus avec les modèles BM25 et TF-IDF sont encore très proches.

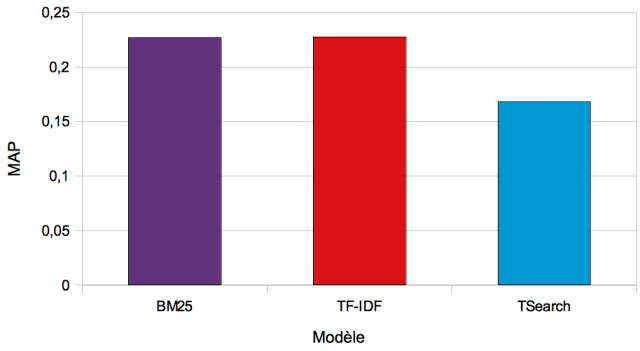


FIGURE 24 – Comparaison de la MAP des modèles TSearch, TF-IDF et BM25 pour le corpus TREC 8

Ces résultats confirment nos intuitions concernant les performances de TSearch inférieures aux approches de la littérature. L'utilisation d'un modèle alternatif constitue donc un premier axe d'amélioration.

5.4.1.4 Vers une solution alternative à TSearch : Apache Solr

Pour apporter une réponse aux lacunes de TSearch, nous avons mené un comparatif de solutions open source alternatives disponibles sur le marché. Ce comparatif nous a conduit à retenir le moteur de recherche Apache Solr [1]. Les principaux arguments en sa faveur sont :

- sa licence open source autorisant l'adaptation de la solution à nos besoins ;
- son coût limité de mise en œuvre ;
- la présence d'une communauté importante et active garantissant le support et la pérennité de l'outil ;
- des performances reconnues par une vaste communauté et proches des modèles de la littérature ;
- la maturité du projet ;
- et son déploiement sous la forme d'un service externe qui facilite grandement son intégration dans une architecture existante.

Solr a également été comparé aux autres approches sur des collections de référence TREC (cf figure 25). Solr a donc été configuré pour suivre le même

1. http ://lucene.apache.org/solr/

processus d'indexation que TSearch sur la plateforme *OverBlog*. Ses bonnes performances ont également permis d'indexer des caractéristiques supplémentaires des documents et de s'affranchir de l'heuristique complexe utilisée par TSearch pour la recherche par trimestre. Afin d'exploiter les caractéristiques des blogs, l'ordonnancement des résultats tient compte à la fois de l'importance des différentes éléments indexés (titre, corps, tags, ...) et de la fraîcheur des documents, c'est-à-dire de leur date de publication.

5.4.1.5 Bilan un mois après la migration de TSearch à Solr

Afin de montrer l'amélioration des performances induite par le passage de TSearch à Solr, nous avons procédé à une évaluation de deux mois (du 7 octobre 2010 au 6 décembre 2010) durant lesquels nous avons utilisé successivement les deux solutions. Ainsi, l'ensemble des requêtes du premier mois ont été soumises à TSearch, et celles du second mois (à partir du 7 novembre 2010) ont été traitées par Solr. Les deux moteurs de recherche ont fonctionné parallèlement afin de pouvoir passer de l'un à l'autre au besoin.

L'évaluation repose sur :

— les performances basées sur des métriques de recherche d'information ;
— les performances opérationnelles évaluées au travers des temps de réponse et des taux de clics.

Une pertinence accrue des résultats Similairement à l'évaluation de TSearch, nous avons comparé le modèle de recherche de Solr au modèle TF-IDF sur la collection TREC 8. Ce choix a été motivé par le fait que le modèle de Solr dérive de TF-IDF mais n'est pas identique à l'approche implantée dans Terrier, et que les performances entre BM25 et TF-IDF sont quasiment identiques.

La figure 25 présente les courbes Rappel/Précision des trois modèles comparés. Nous constatons que les courbes de TF-IDF et de Solr sont très proches. Les performances du modèle TSearch demeurent inférieures à celles des autres modèles.

Ces résultats se confirment dans la figure 26. Nous notons en effet que la MAP suit la même tendance. TF-IDF et Solr sont proches et dépassent TSearch.

De meilleurs temps de réponse La figure 27 présente la proportion de requêtes traitées en dessous d'un certain temps selon la légende suivante :

— Vert clair : moins de 200ms (millisecondes) ;
— Vert foncé : entre 200ms et 1000ms ;
— Bleu : entre 1000ms et 2500ms ;
— Rouge : entre 2500ms et 10000ms ;

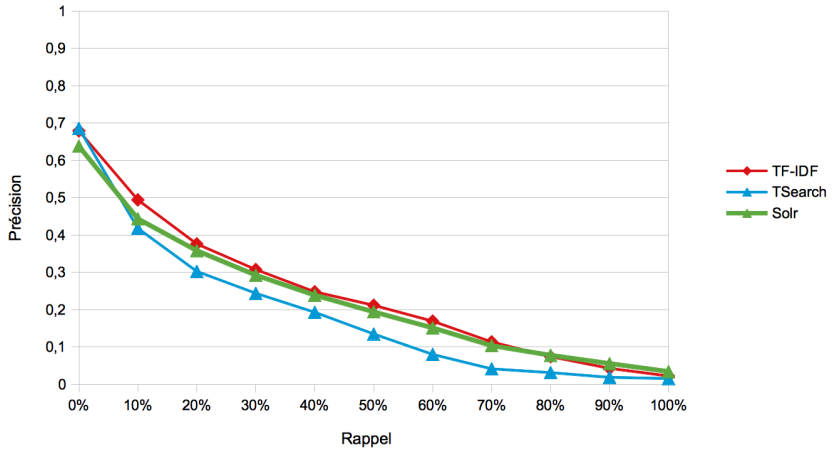


FIGURE 25 – Comparaison des courbes Rappel/Précision des modèles TSearch, Solr et TF-IDF pour le corpus TREC 8

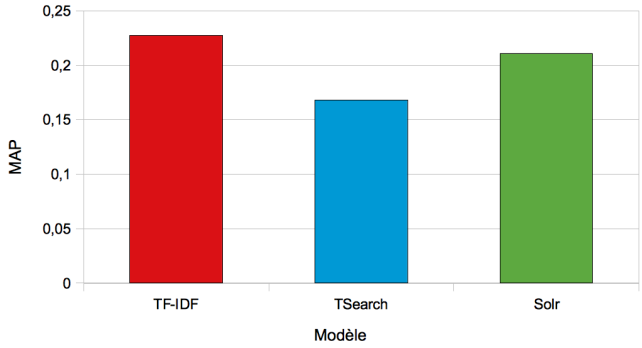


FIGURE 26 – Comparaison de la MAP des modèles TSearch, Solr et TF-IDF pour le corpus TREC 8

— Noir : plus de 10000ms.

Nous constatons que le passage à Solr conduit à une amélioration importante des temps de réponse. La proportion de requêtes répondant en moins de 200ms représente plus de 10% des requêtes contre moins de 1% avec TSearch. 70% des requêtes sont traitées en moins de 1000ms et 90% en moins de 2000ms. Enfin, les requêtes supérieures à 10000ms ont été complètement éliminées.

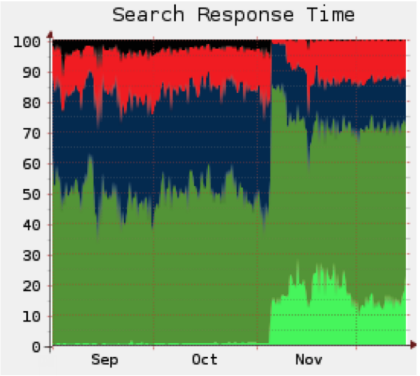


FIGURE 27 – Temps de réponse du moteur de recherche avant et après la migration à Solr

Une utilisation accrue de la part des visiteurs L'amélioration des performances se traduit également par une utilisation accrue du moteur de recherche suite à la migration, comme le montre le tableau 7. Bien que la croissance du nombre de recherches uniques soit modérée (+6,8%), et que cette croissance pourrait être une conséquence de l'augmentation normale du trafic de la plateforme, nous constatons une forte augmentation du nombre de clics générés (+31%). Le taux de clics connaît par conséquent une croissance de près de 26%.

Métrique	**Moteur de recherche**	**Valeur**	**Écart**
Recherches uniques	TSearch	36668	
	Solr	39343	+6,8%
Clics générés	TSearch	8147	
	Solr	11803	+31%
Taux de clics	TSearch	22,2%	
	Solr	30%	+26%

TABLE 7 – Statistiques d'utilisation du moteur de recherche d'*OverBlog* avant et après la migration de TSearch à Solr

5.4.1.6 Conclusions sur l'amélioration du moteur de recherche

Les différents indicateurs mis en place ont montré une amélioration des performances du moteur de recherche à plusieurs niveaux :

— l'amélioration du taux de clics et de la MAP traduit une pertinence des documents restitués accrue ;
— les temps de réponse ont quant à eux été réduits et rendent l'utilisation des mesures d'intérêts possibles ;

Les résultats détaillés de cette évaluation et du bilan de la migration font l'objet d'un rapport de recherche (Dudognon, 2010).

Le second axe envisagé pour garantir une meilleure qualité des recommandations est la mise en place d'outils de lutte contre les splogs et de critères de qualité. Les travaux menés en ce sens sont présentés dans la section suivante.

5.4.2 Qualité des données et lutte contre les splogs

La sélection des documents doit garantir un certain niveau de qualité afin de proposer aux visiteurs des recommandations qui les amèneront à avoir confiance en l'outil et à l'adopter (Sarwar *et al.*, 2000a). L'aspect qualitatif des recommandations est dégradé par la présence de splogs. Il est également conditionné par des critères "esthétiques" régissant la présentation des recommandations.

5.4.2.1 Détecter et supprimer les splogs

La détection et l'élimination des splogs s'effectuent à deux niveaux :

— lors de la création d'un nouveau blog sur la plateforme ;
— et lors de la publication de nouveaux billets.

La première solution est une étape préventive visant à éviter la création de splogs. La seconde solution est curative et consiste à éliminer des splogs déjà présents sur la plateforme.

Prévenir plutôt que guérir La prévention est la moins coûteuse des deux options. Elle permet en effet d'éviter la consommation de ressources (espace disque, consommation mémoire, temps de calcul, ...) par les splogs. Deux stratégies ont donc été mises en place pour prévenir la création de splogs.

La première stratégie repose sur un service de prédiction permettant de déterminer si un blog est potentiellement un splog. Cette prédiction se base sur l'analyse de l'adresse de courriel et du nom de domaine utilisés lors de l'inscription à la plateforme, ainsi que sur une base d'apprentissage contenant des blogs

"légitimes" et des splogs. Le modèle utilisé est le modèle C4.5 proposé par Quinlan (1993).

A chaque inscription, le service de prédiction est appelé et retourne une note de confiance. Si cette note est en dessous d'un certain seuil alors le formulaire est complété par un CAPTCHA (*Completely Automated Public Turing test to Tell Computers and Humans Apart*) (Von Ahn *et al.*, 2003), c'est-à-dire d'un test permettant de distinguer un usager humain d'un programme informatique. L'inscription ne peut être validée que si le résultat du test est positif.

La seconde stratégie consiste à interdire la publication aux usagers n'ayant pas validé leur compte par courriel. Suite à l'inscription, un courriel comportant un lien hypertexte est envoyé à l'adresse renseignée dans le formulaire d'inscription. Ce lien hypertexte conduit à la validation du compte de l'usager et permet donc d'autoriser la publication de contenus.

La collection a cependant été polluée avant la mise en place de ces deux stratégies. Une phase curative est donc nécessaire pour éliminer les splogs déjà présents.

La suppression des splogs existants A partir des mêmes données d'apprentissage que celles utilisées pour le service de prédiction à la création, nous avons construit un modèle de prédiction reposant sur les caractéristiques des blogs de cette base d'apprentissage. Plusieurs indicateurs ont été définis pour caractériser ces blogs, comme par exemple le nombre de billets rédigés, le nombre moyen de liens hypertexte par billet, la longueur des billets ou encore la période d'activité du blog.

Une fois le modèle de prédiction établi, nous avons pu procéder à l'analyse de l'ensemble de la collection. Un indice de confiance ainsi qu'une étiquette ("splog" ou "valide") ont été attribués à chacun des blogs de la plateforme. Les blogs jugés "valides" ont été ignorés. Les blogs étiquetés comme "splogs" et ayant un indice de confiance élevé ont été éliminés automatiquement. Les blogs restants ont quant à eux été validés manuellement.

5.4.2.2 Des critères de qualité pour sélectionner les contenus

Un ensemble de critères, auxquels doivent répondre les billets recommandés, a été fixé arbitrairement par l'équipe en charge de la définition des besoins et des spécifications du produit. Il s'agit par exemple d'un nombre minimal de caractères pour le titre et le corps du billet, de la présence d'une image de qualité suffisante pour générer une vignette, ou du classement du blog selon une formule de "Blog Rank" propre à la plateforme *OverBlog* (qui tient compte de l'activité du blogueur,

de sa popularité, de son influence, ...).

FIGURE 28 – Exemple de billet de blog pour lequel des recommandations sont proposées

L'objectif premier de ces critères est de garantir l'esthétique de l'affichage des recommandations, et ce en respectant la charte graphique du blog sur lequel elles apparaissent. Ainsi, pour le billet illustré en figure 28, les recommandations seront proposées aux visiteurs sous la forme d'un pavé additionnel faisant suite au billet. Ce pavé est présenté en figure 29.

5.4.3 Mesures de sélection utilisées

Compte tenu des performances mesurées lors de l'expérience utilisateur et des contraintes industrielles imposées, les mesures *blog*, *mlt* et *search* ont été retenues à l'issue de la phase de préparation des données pour l'implantation sur *OverBlog*. Cette phase de préparation des données a conduit à la réduction de moitié de la taille de la collection utilisée pour générer des recommandations, ramenant sa taille à 12 millions de billets. Les mesures *topcateg* et *kmeans* ont été éliminées car non viables sur le système en ligne en raison d'un temps de calcul trop important.

Afin de faire valider notre système de recommandation par l'équipe technique, nous avons réalisé un test de performance simulant le calcul des recommandations pour 20 billets en parallèle et pour un total de 10000 billets.

FIGURE 29 – Exemple de bloc de recommandations présenté aux visiteurs

Temps de réponse (en ms)	**Nombre de requêtes**	**Proportion cumulée**
inférieur à 100ms	122	1.22%
entre 100ms et 200ms	305	4.27%
entre 200ms et 300ms	401	8.28%
entre 300ms et 400ms	911	17.39%
entre 400ms et 500ms	1365	31.04%
entre 500ms et 600ms	1505	46.09%
entre 600ms et 700ms	1350	59.59%
entre 700ms et 800ms	1083	70.42%
entre 800ms et 900ms	880	79.22%
entre 900ms et 1000ms	626	85.48%
entre 1000ms et 2000ms	1340	98.88%
supérieur à 2000ms	112	100%

TABLE 8 – Répartition des requêtes au service de recommandation en fonction des temps de réponse

Le temps de calcul moyen relevé lors du test est de 705ms. Le tableau 8 présente le détail des résultats obtenus. Nous constatons que 85,48% des requêtes sont

traités en moins de 1000ms (c'est-à-dire sous le seuil initialement fixé) et que 98,88% d'entre elles le sont en moins de 2000ms. Cet écart qui concerne moins de 15% des requêtes a été jugé acceptable par l'équipe technique. L'intégration du système de recommandation au produit a donc été validée.

5.5 Architecture du système de recommandation implanté

Conçu comme un service de recommandation externe et autonome, afin de limiter l'impact sur le produit existant, le système de recommandation est interrogé par la plateforme de blogs de manière asynchrone à l'aide d'un appel JavaScript.

Cette section a pour objectif de présenter l'infrastructure globale du système de recommandation ainsi que son intégration au sein du produit existant. Elle se focalise ensuite plus en détail sur les interactions entre les différents composants du système de recommandation, pour finalement en définir la structure.

La plateforme *OverBlog* suit un modèle d'architecture multi-tiers, composé de trois couches distinctes :

- la couche *présentation* qui correspond à la partie visible de l'application et gère les interactions avec les usagers ;
- la couche *métier* implémentant la logique métier de l'application. Elle offre des services à la couche *présentation* ;
- et la couche d'*accès aux données* qui fournit de manière transparente à la couche *métier* des données propres aux systèmes ou issues de services externes.

Notre modèle se situe au niveau de la couche *métier*. Il assure également une partie de l'accès aux données stockées. La figure 30 illustre cette intégration sous la forme d'un service de recommandation.

Lorsqu'un usager accède à un billet, la couche présentation appelle de manière asynchrone, et parallèlement à l'accès aux autres services de la plateforme *OverBlog*, le service de recommandation. Cet appel est associé à un délai d'expiration au-delà duquel il sera abandonné. En cas d'expiration, aucune recommandation n'est affichée. Le calcul est cependant maintenu afin d'être en mesure de fournir des recommandations lors du prochain appel.

Lors du premier appel au service de recommandation, la demande est transmise à l'agrégateur qui se charge d'interroger les différentes mesures de sélection disponibles. Chaque mesure extrait de la collection, en utilisant d'éventuelles sources d'information complémentaires (par exemple des statistiques), les documents répondant à un intérêt particulier et leur associe un score. Les ensembles de couples

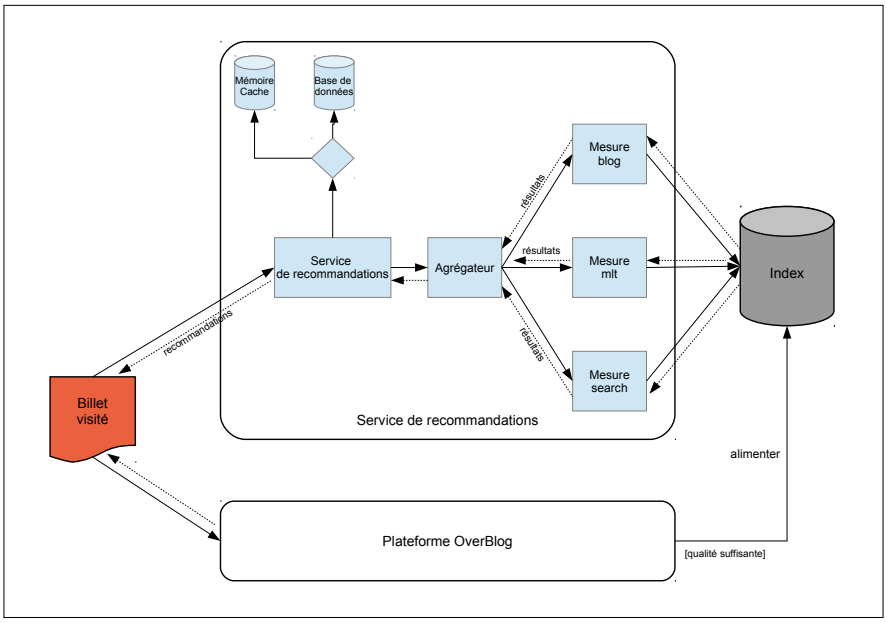


FIGURE 30 – Schéma global de l'architecture du système de recommandation implanté sur la plateforme *OverBlog*

document/score restitués par les mesures de sélection sont ensuite agrégés pour produire une liste unique de recommandations. Chaque recommandation peut provenir d'une ou plusieurs mesures de sélection. L'agrégateur transmet enfin cette liste de recommandations au service de recommandation qui les sauvegarde de manière persistante en base de données. La liste est également placée pour une durée de 24 heures dans une mémoire cache pour accélérer les accès aux données.

Lorsque des recommandations sont demandées pour un billet donné, le service de recommandation vérifie d'abord s'il dispose d'éléments dans la mémoire cache. Si aucun élément n'est trouvé, il interroge la base de données. Enfin, en l'absence de recommandations en mémoire cache et en base de données, la demande de calcul est transmise à l'agrégateur.

5.6 Évaluation du système de recommandation intégré à *OverBlog*

Cette évaluation porte sur l'implantation du modèle proposé sur la plateforme de blogs *OverBlog*. L'objectif est de vérifier, dans un contexte réel d'utilisation,

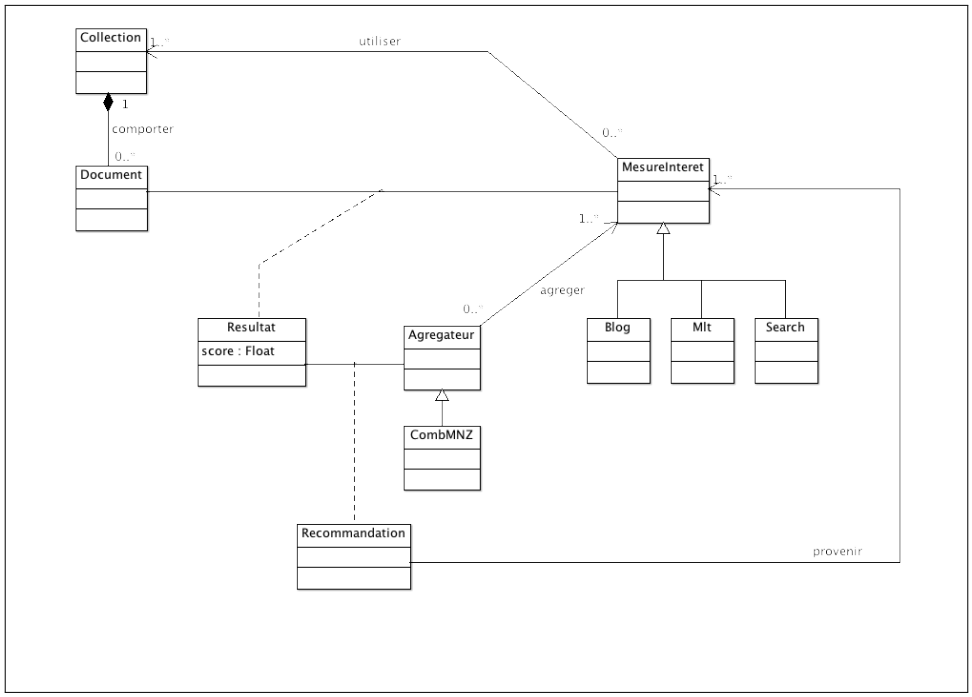


FIGURE 31 – Diagramme de classes du service de recommandation

l'intérêt des recommandations fournies par notre modèle par rapport à des mesures considérées indépendamment. Nous souhaitons également nous assurer que notre modèle est compatible avec un passage à l'échelle.

Après avoir explicité le protocole d'évaluation mis en œuvre, nous présentons les résultats obtenus sur *OverBlog*. L'évaluation se focalise sur l'usage des recommandations par les usagers, ainsi que sur la viabilité du modèle d'un point de vue économique, c'est-à-dire pour les éditeurs de plateforme de blogs.

5.6.1 Protocole et métriques utilisés

5.6.1.1 Protocole d'évaluation

Pour cette expérimentation, nous avons comparé les résultats de notre modèle aux mesures de sélection déjà intégrées à la plateforme *OverBlog*. Il s'agit des trois mesures suivantes (dont les deux premières sont des variantes de mesures de contenu) :

— **MoreLikeThis (*mlt*)** : recherche dans la collection les billets de blogs possédant également les termes les plus représentatifs du billet visité. Le contenu global des billets est analysé ;
— **SearchSim (*search*)** : repose sur le contenu textuel des titres des billets de blogs uniquement. Cette mesure favorise les billets de blogs ayant des termes similaires aux termes du billet visité ;
— **BlogArticle (*blog*)** : retourne des billets appartenant au même blog que le billet visité. Cette mesure ne tient pas du tout compte du contenu textuel.

Outre ces mesures, nous avons intégré une version simplifiée du modèle d'agrégation (*fused*) présenté dans la section 4.5 pour construire la liste de recommandations finale. Le choix d'une version simplifiée a été motivé par le fait que le modèle est intégré à une plateforme en exploitation. La complexité de calcul doit donc être limitée et maîtrisée pour ne pas mettre en péril l'ensemble de la plateforme. Le modèle d'agrégation repose sur la sélection, pour chacune des trois mesures de sélection précédentes, des deux recommandations ayant le meilleur score. En assurant la représentativité de chaque mesure de sélection, le résultat de cette agrégation offre une diversité des recommandations, et ce sur différentes dimensions (contenu, appartenance à un blog).

Les recommandations, au nombre de six, sont présentées dans un bloc positionné à la suite d'un billet (cf. figure 29) et la mesure utilisée pour les calculer (*blog*, *mlt*, *search* ou *fused*) est choisie aléatoirement pour chaque billet. Les recommandations sont ensuite figées pour chaque billet qui se trouve finalement associé à une et une seule mesure.

Dans le cas de la mesure agrégée *fused*, quatre recommandations provenant de chaque mesure d'intérêts sont extraites. La liste finale présente également six recommandations en sélectionnant deux éléments de chaque mesure. Les éléments non utilisés serviront lors de la phase d'apprentissage.

Quelle que soit la mesure utilisée, la liste est présentée dans un ordre aléatoire afin d'éliminer le biais lié à la position des recommandations. En effet, les premiers résultats sont généralement ceux qui rassemblent le plus grand nombre de clics. Ceci s'est confirmé dans le cas de notre expérimentation où nous avons constaté que les deux premières recommandations de la liste totalisaient un nombre de clics plus important. Ce constat est illustré dans la figure 32 qui montre la répartition moyenne des clics en fonction de la position des recommandations dans la liste.

Pour l'apprentissage, seuls les billets associés à la mesure *fused* ont été utilisés. Il s'agit en effet des seuls billets disposant de recommandations issues de chaque mesure de sélection, et permettant ainsi l'ajustement des proportions. Lors de cet ajustement, la liste finale contient au minimum une recommandation de chaque mesure. Les trois recommandations restantes sont sélectionnées en fonction des taux de clics précédemment enregistrés pour le billet.

L'expérimentation s'est décomposée en trois périodes :

Période 1 Pendant un mois, nous avons comparé notre mesure diversifiée (*fused*) aux trois autres mesures de sélection considérées individuellement. L'objectif est d'évaluer l'intérêt de l'agrégation pour la diversité et la satisfaction des usagers.

Période 2 Cette seconde période de quinze jours vise les mêmes objectifs que la période 1 pour consolider les résultats obtenus.

Période 3 Durant un mois, seule la mesure diversifiée est proposée. Les proportions de recommandations issues de chaque mesure ont été ajustées suite à une phase d'apprentissage exploitant les taux de clics, selon la méthode présentée en section 4.6.

Les métriques utilisées pour l'évaluation sont présentées dans la section suivante.

5.6.1.2 Métriques

Deux types de métriques ont été utilisés au cours de l'expérimentation. Un premier ensemble d'indicateurs a pour objectif de contrôler les éventuels biais identifiés, alors que le second groupe se focalise sur les performances des mesures de sélection.

Contrôler les biais Deux biais majeurs ont été identifiés. Le premier concerne la présentation des résultats : en effet, les premiers résultats de la liste sont ceux qui

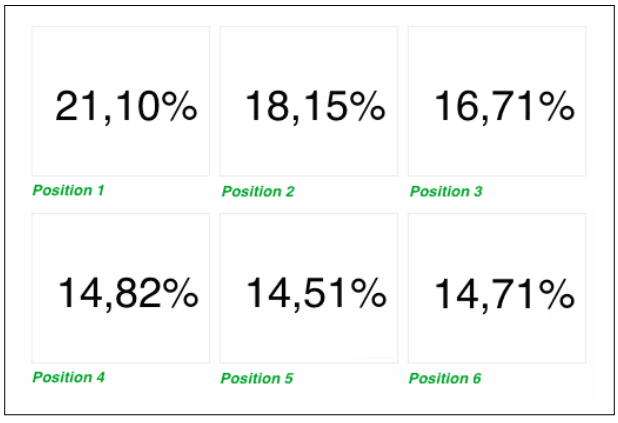


FIGURE 32 – Répartition des clics en fonction de la position des recommandations (le format de six recommandations est imposé par la société OverBlog)

totalisent le plus de clics (Craswell *et al.*, 2008). Cette influence de la position des recommandations sur le taux de clics s'est effectivement confirmée lors de notre expérimentation, comme le montre la figure 32. Ceci justifie l'ordonnancement aléatoire des recommandations à chaque affichage afin de ne pas favoriser une mesure plutôt qu'une autre.

Le second biais concerne la répartition des affichages de manière équitable entre les quatre mesures comparées (*fused*, *blog*, *mlt* et *search*). Le tableau 9 prouve que les quatre mesures ont été proposées aux usagers dans des proportions analogues.

Mesure	**Période 1**	**Période 2**
blog	23,76%	24,14%
mlt	24,14%	23,61%
search	26,30%	27,16%
fused	25,80%	25,09%

TABLE 9 – Répartition des affichages en fonction des mesures de sélection

Déterminer les performances des mesures de sélection Afin d'évaluer les performances de notre modèle et des différences mesures de sélection agrégées, nous avons employé le taux de clics sur les recommandations proposées. Le taux de clics est une approximation de la pertinence perçue par les usagers souvent retenue pour l'évaluation de systèmes en ligne, c'est-à-dire sans collection de référence (Joachims *et al.*, 2005) (Chapelle et Zhang, 2009) (Guo *et al.*, 2009).

Ce taux de clics a été mesuré à trois niveaux :
— globalement, c'est-à-dire indépendamment des mesures utilisées ;
— pour chaque mesure de sélection considérée individuellement ;
— et au niveau de la mesure *fused* afin de déterminer la provenance des recommandations cliquées.

La section suivante présente les taux de clics obtenus au cours des différentes phases de l'expérimentation et permettant de justifier notre proposition dans un contexte industriel.

5.6.2 Résultats de l'évaluation

L'évaluation du système de recommandation s'étend sur une période de deux mois et demi :
— un mois et demi a été consacré à l'évaluation de l'agrégation ;

— le mois restant a, quant à lui, été dédié à l'évaluation du processus d'apprentissage.

5.6.2.1 L'évaluation de la phase d'agrégation

Dans le but de démontrer l'apport de l'agrégation de mesures de sélection de natures différentes, nous nous sommes dans un premier temps focalisés uniquement sur la comparaison des mesures considérées individuellement avec notre mesure agrégée *fused*. Pour cela, les quatre mesures ont été utilisées parallèlement au cours d'une période d'un mois, puis lors d'une seconde période de quinze jours.

Durant la première période (août 2012), le bloc de recommandations a été affiché 886 041 fois. Le tableau 10 précise le taux de clics moyen relevé pour chacune des mesures utilisées. Nous notons qu'en moyenne, et indépendamment des mesures utilisées, le taux de clics a été de 3.89%. Nous constatons également que les trois mesures de la plateforme *OverBlog* ont un taux de clics inférieur à 4% alors que notre modèle conduit à un taux de clics supérieur (4,70%). Ceci démontre l'intérêt de l'approche d'agrégation par rapport aux mesures considérées individuellement.

Mesure	**Taux de clics**
mlt	3,55%
search	3,95%
blog	3,49%
fused	4,70%
Moyenne	3,89%

TABLE 10 – Taux de clics moyen par mesure de sélection lors de la première période de l'évaluation de l'agrégation

La figure 33 présente quant à elle l'évolution des taux de clics par mesure au cours de cette même période. Elle confirme que notre modèle conduit à un meilleur taux de clics tout au long de l'expérimentation.

Bien que les principes sur lesquels elles reposent soient très différents, les mesures de sélection utilisées obtiennent des taux de clics relativement proches (par exemple 3,49% pour *blog* et 3,55% pour *mlt*). Ce constat vient appuyer les résultats de l'étude utilisateur présentés en section 4.2. Ils confirment en effet que les intérêts des usagers ne sont pas nécessairement liés à la proximité thématique

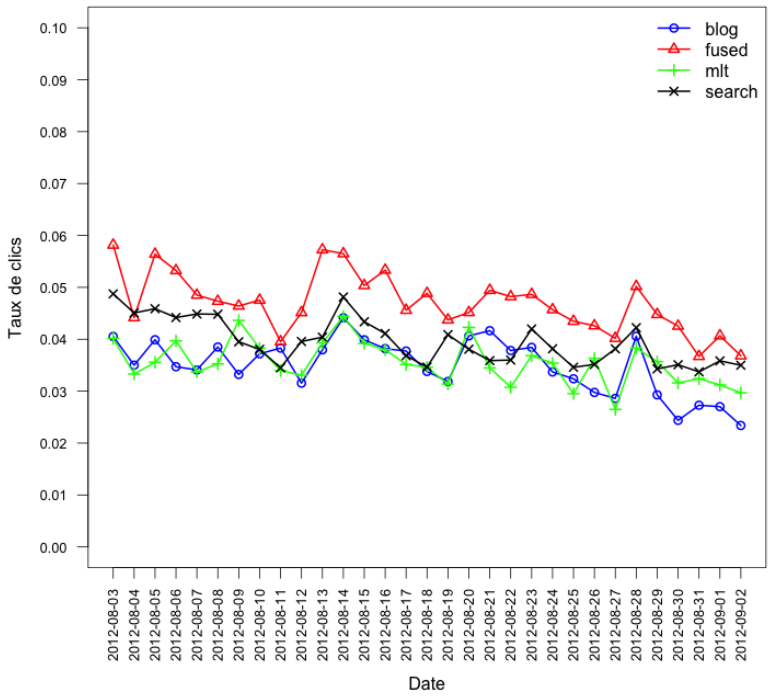


FIGURE 33 – Taux de clics quotidien au cours de la première période d'évaluation (août 2012)

des recommandations, et qu'ils peuvent être satisfaits par des recommandations diversifiées.

Afin de consolider les résultats obtenus, nous nous sommes assurés qu'ils se vérifiaient sur une seconde période de 15 jours (première quinzaine du mois de septembre 2012), pendant laquelle le bloc de recommandations a été présenté 584 884 fois. Le tableau 11 et la figure 34 confirme les observations faites lors de la première période : les taux de clics des mesures *blog*, *mlt* et *search* demeurent en dessous de 4%, et notre modèle *fused* conduit à un taux de clics supérieur (4,1%). Ceci se vérifie sur l'ensemble de la seconde période.

Entre les deux périodes d'évaluation, nous notons une baisse globale du taux de clics et ce quelle que soit la mesure considérée. Le fait que les recommandations soient pré-calculées, et qu'elles soient donc identiques d'une visite sur l'autre, peut expliquer que l'intérêt qu'elles suscitent pour un visiteur récurrent sera réduit. L'expérimentation étant réalisée sur une plateforme française, nous pourrions penser que la période de vacances a eu une influence sur le temps passé sur la plateforme, ce qui pourrait également expliquer la baisse du taux de clics. La

Mesure	Taux de clics
mlt	3,20%
search	3,77%
blog	2,72%
fused	4,10%
Moyenne	3,46%

TABLE 11 – Taux de clics moyen par mesure de sélection lors de la seconde période de l'évaluation de l'agrégation (septembre 2012)

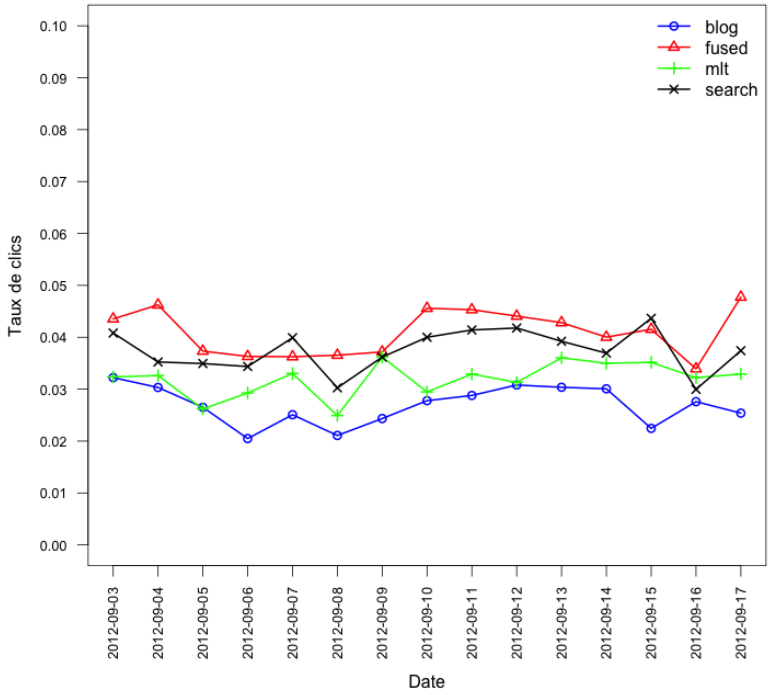


FIGURE 34 – Taux de clics quotidien au cours de la seconde période d'évaluation de l'agrégation (du 3 au 18 septembre 2012)

figure 35 démontre que la durée des sessions utilisateur a été à peu près identique au cours des deux périodes et permet par conséquent d'éliminer cette seconde hypothèse.

L'intérêt de l'agrégation se traduit également au niveau de la répartition des

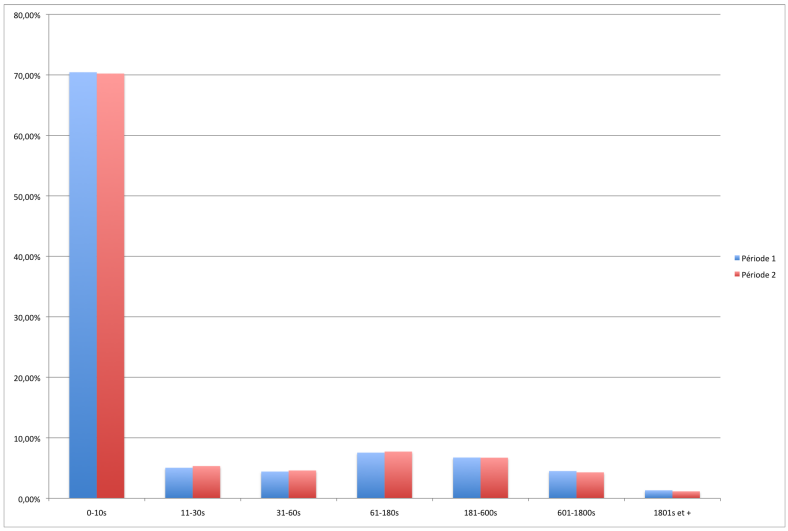


FIGURE 35 – Comparaison de la durée des sessions utilisateurs au cours des deux périodes d'évaluation de l'agrégation

clics entre les recommandations constitutives de la liste *fused* qui sélectionne les deux meilleurs résultats de chacune des trois autres mesures. Le tableau 12 illustre la provenance des recommandations dans la mesure *fused*. Il montre l'importante variété des recommandations, notamment au travers de la présence non négligeable de la mesure *blog* (24.02% des clics en moyenne lors de la première période). Nous constatons dans le même temps que les mesures de contenu, et plus particulièrement *search* qui utilise les mots du titre, ont un impact plus important. Ces résultats sont valables pour les deux périodes de l'évaluation.

Mesure	**Proportions**	
	Période 1	**Période 2**
blog	24.02%	24,08%
mlt	28.00%	28,66%
search	47.98%	47,26%

TABLE 12 – Provenance des recommandations cliquées dans la mesure *fused* lors de l'évaluation de l'agrégation

Chaque billet visité pouvant susciter des intérêts différents, l'intégration d'un processus d'apprentissage est justifiée car elle permet de moduler l'agrégation des

mesures de sélection de façon plus précise. C'est ce que nous avons souhaité vérifier au travers de l'évaluation de la phase d'apprentissage.

5.6.2.2 L'apprentissage améliore les performances

L'apprentissage a pour objectif de conduire à des recommandations plus proches des intérêts réels des usagers. Pour évaluer son apport, nous avons considéré les clics comme des jugements d'intérêt, et nous les avons utilisés pour favoriser, ou au contraire pénaliser, certaines mesures de sélection au moment de l'agrégation.

Suite à la phase d'apprentissage, les trois mesures ont été agrégées de la manière suivante :

— une recommandation est extraite de chaque mesure de sélection (celle ayant le meilleur score). Nous obtenons alors une première sélection de trois recommandations où toutes les mesures sont représentées ;
— les trois recommandations manquantes sont ensuite sélectionnées en fonction du taux de clics des mesures de sélection pour le billet visité.

Si nous supposons un billet ayant totalisé six clics sur les recommandations provenant de la mesure *blog*, trois sur celles de la mesure *mlt*, et aucun pour la mesure *search*, et en respectant les principes évoqués précédemment, la liste finale comportera :

— trois recommandations issues de la mesure *blog* ;
— deux recommandations issues de la mesure *mlt* ;
— et une recommandation issue de la mesure *search*.

Les listes de recommandations ont donc été recalculées en intégrant les données issues de l'apprentissage. L'évaluation a ensuite été menée durant une période d'un mois où seule la mesure *fused* a été présentée aux visiteurs de la plateforme. Le taux de clics moyen relevé au cours de cette période est de 5,43%. Cette valeur est nettement supérieure au taux de 4,7% constaté lors de l'évaluation de l'agrégation. La figure 36 montre l'évolution de ce taux de clics au cours de l'expérimentation. Nous observons la même tendance que lors de l'évaluation de l'agrégation : le taux de clics baisse progressivement et s'explique de notre point de vue essentiellement par le fait que les recommandations sont figées pour toute la durée de l'expérimentation. Leur intérêt est là encore réduit pour les visiteurs ayant déjà vu le billet.

Le tableau 13 présente la répartition des clics en fonction des mesures d'intérêts agrégées. Nous constatons que le taux de clics de la mesure *search* s'est maintenu, et que celui de la mesure *blog* a légèrement baissé au profit de celui de la mesure *mlt*. Le taux de clics de la mesure *blog* reste cependant au dessus de 20%. Cette mesure apporte une part non négligeable de recommandations pertinentes qui n'auraient

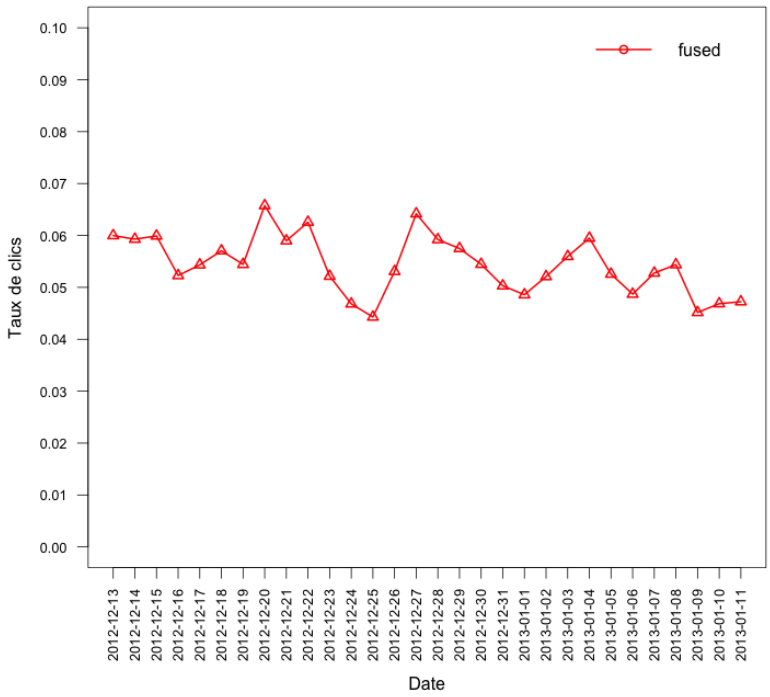


FIGURE 36 – Taux de clics quotidien lors de l'évaluation de l'apprentissage

pas été restituées en utilisant uniquement des mesures reposant sur le contenu. Notre liste diversifiée est donc à même de satisfaire un panel plus large d'intérêts. Ces résultats s'avèrent à nouveau conformes aux attentes évoquées à l'issue de l'étude utilisateur présentée en section 4.2. Ils valident notre modèle en termes de satisfaction des usagers dans un contexte industriel.

Il convient à présent de s'assurer de la viabilité de notre proposition pour un éditeur de plateforme de blogs, c'est-à-dire du point de vue des performances opérationnelles, du passage à l'échelle et du modèle économique.

5.6.2.3 Viabilité de la proposition d'un point de vue industriel

Volumétrie et passage à l'échelle L'expérimentation a été réalisée dans un contexte réel qui implique une volumétrie conséquente. La collection d'où ont été extraites les recommandations comporte environ vingt millions de billets.

Les recommandations n'ont été calculées que durant les périodes 1 et 2 de l'expérimentation qui s'étendent sur une durée totale d'un mois et demi. Elles sont associées à 632745 billets différents répartis sur 84012 blogs qualifiés de "délaissés",

Mesure	Proportions
blog	20,89%
mlt	31,59%
search	47,52%

TABLE 13 – Provenance des recommandations cliquées dans la mesure *fused* lors de l'évaluation de l'apprentissage

c'est-à-dire sans activité (publication, connexion à l'interface d'administration, ...) du blogueur depuis au moins 45 jours (cette durée d'inactivité est fixée arbitrairement par l'équipe chargée des spécifications de la plateforme *OverBlog*). Sur cette même période, les blocs de recommandations ont été proposés près de 1,5 millions de fois aux visiteurs.

Malgré ces volumes de données conséquents, le système de recommandation implanté garantit un temps de réponse inférieur à la seconde et est ainsi viable sur le plan opérationnel et à grande échelle.

Modèle économique Notre modèle se justifie également sur le plan économique. En effet, un taux de clics de 5,63% est jugé comme important. En considérant le trafic mensuel de la plateforme *OverBlog*, estimé à 34 millions de pages vues [1], ce taux de clics conduirait à une augmentation du trafic de près de deux millions de pages vues par mois. Une fois monétisée, ceci se traduirait par une augmentation des revenus publicitaires de *OverBlog* estimée à plus de 110000€ par an.

1. ComScore, http ://www.comscore.com, décembre 2012

Conclusion et perspectives

Les usagers ont des attentes différentes vis-à-vis de l'information. Tout système qui vise à apporter des réponses pertinentes à leurs besoins doit prendre en compte cet état de fait. Les systèmes de recherche d'information ainsi que les systèmes de recommandation, qu'ils emploient des approches basées sur le contenu ou le filtrage collaboratif, ont cette ambition. Pour répondre au mieux à cette pluralité d'intérêts, la communauté s'est orientée vers la notion de diversité. Il s'agit de proposer des informations variées plutôt que similaires. Nos travaux se sont ainsi orientés vers la prise en compte de la diversité au sein des systèmes de recommandation dans une dimension théorique par la définition d'un modèle, et une dimension pratique par son évaluation et son intégration dans la plateforme d'hébergement de blogs *Overblog*, offre de *Ebuzzing Group*, groupe leader européen du média social.

Cette thèse propose une nouvelle approche qui offre une solution au problème du démarrage à froid et au sur-apprentissage. En effet, les systèmes de recommandation usuels ont tendance à souffrir du sur-apprentissage qui conduit à proposer toujours les mêmes items aux utilisateurs. Par ailleurs, il leur est également difficile de recommander des items nouveaux et/ou des items à de nouveaux utilisateurs pour lesquels les profils sont inconnus.

Notre approche s'appuie sur la diversité des recommandations pour résoudre ces problèmes en suggérant, par des mesures de sélection particulières, des items qui n'auraient pas été suggérés par un filtrage collaboratif ou par des approches basées uniquement sur le contenu.

Nous ne posons aucune hypothèse préalable sur les intérêts de l'usager et nous ne considérons que le seul document visité comme référence pour la phase de recommandation. La variété des intérêts que ce document peut susciter est traduite par l'emploi de plusieurs mesures de sélection. Ces dernières donnent lieu, après agrégation, à une liste unique et diversifiée de recommandations. Cette diversité permet de maximiser la satisfaction de l'usager.

Par ailleurs, notre approche s'appuie également sur un rafraîchissement récurrent des connaissances stockées (informations sur les parcours d'usagers,

sur les contenus, sur la pertinence des items recommandés inférée par les clics) qui permet une pondération adaptative des contributions de chaque mesure de sélection dans les recommandations et leur ordonnancement.

En complément du modèle proposé, cette thèse se concrétise à travers plusieurs contributions, tant d'un point de vue théorique, que d'un point de vue pratique et industriel.

Une première étude inspirée des travaux de Lee (1997) nous a tout d'abord permis de valider les hypothèses sur lesquelles repose notre modèle. Cette étude a été menée sur la collection TREC 3, mais également sur la collection de référence TREC Web 2009, plus représentative du Web. Nous avons ainsi démontré que :

— différentes mesures de sélection conduisent à des documents pertinents différents, et ce même lorsqu'elles s'attachent à satisfaire un objectif commun ;
— l'agrégation de ces mesures améliore la précision ;
— l'agrégation de mesures différentes impacte positivement la diversité des recommandations.

Nous avons également souligné le fait que les documents pertinents sont majoritairement localisés en tête de liste, et qu'il est par conséquent tout à fait légitime de se limiter à l'agrégation de listes de résultats réduites, par exemple en ne considérant que les 10 premiers documents.

Nous avons par ailleurs évalué notre modèle au travers de deux évaluations.

La première se matérialise par une expérience utilisateur. Cette évaluation a permis de montrer que notre approche répond à un panel d'intérêts plus large, tandis que les autres approches se focalisent sur les intérêts de la majorité des utilisateurs. Les mesures traduisant la "sérendipité" produisent des résultats jugés comme pertinents par les usagers, bien qu'éloignés de la thématique du document visité, et qui ne sont restitués par aucune des mesures reposant sur le contenu.

La seconde évaluation a été menée dans un contexte industriel à grande échelle. L'intégration de notre modèle à la plateforme *OverBlog* a permis de valider sa viabilité sur le plan opérationnel, en respectant les contraintes de l'environnement de production. Les taux de clics obtenus avec notre modèle sont également supérieurs à ceux des approches non agrégées et non diversifiées. Le processus d'apprentissage a finalement conduit à l'amélioration des performances de notre modèle. Nous avons évalué monétairement l'apport de notre contribution et avons montré qu'il était significatif. L'approche que nous avons développée au cours de cette thèse a été intégrée au système commercial en ligne.

Les perspectives découlant de nos travaux sont nombreuses. Les premiers travaux à mener portent sur l'amélioration des différents composants du modèle. Notre modèle étant générique, chaque composant peut être redéfini ou étendu.

Les mesures de sélection constituent un premier axe d'amélioration. Nous envisageons tout d'abord d'étendre le modèle en intégrant de nouvelles mesures afin de couvrir une plus grande variété d'intérêts. Pour sélectionner de manière plus pertinente les bonnes mesures, il est nécessaire de définir précisément les typologies d'intérêts des usagers et de tendre vers une classification des mesures de sélection. Certaines mesures sont en effet plutôt généralistes alors que d'autres répondent au contraire à des intérêts très spécifiques. Une meilleure connaissance des intérêts et des mesures associées conduiraient à une meilleure adaptabilité du système. On pourrait ainsi imaginer des groupes de mesures à combiner qui soient différents en fonction du type des utilisateurs, des types d'informations gérées ou de caractéristiques de l'information courante à partir de laquelle est produite la liste de recommandations.

La phase d'agrégation pourrait également être améliorée en considérant des fonctions différentes. Il pourrait s'agir par exemple d'affiner les phases de sélection et de fusion des listes de résultats.

Le dernier composant de notre méthode est le processus d'apprentissage qui repose sur une stratégie d'interprétation des clics relativement simpliste. En effet, chaque clic est interprété indépendamment et est considéré comme un jugement positif. Bien que cette approche présente l'avantage de pouvoir être mise en place facilement et qu'elle permet une amélioration notable des résultats (augmentation de 20% du taux de clics), plusieurs biais ont été identifiés dans la littérature concernant l'usage des clics comme indicateurs de la pertinence. Notre modèle pourrait donc être complété en considérant des approches plus complexes comme celle proposée par Guo *et al.* (2009) qui considèrent les sessions, c'est-à-dire des clics multiples sur une même liste de résultats. D'autres approches, comme (Laporte *et al.*, 2012), montrent également que les clics peuvent traduire des intérêts différents.

Nous avons par ailleurs initié l'expérimentation de notre modèle avec une collection de documents portant sur un autre domaine. Dans ce contexte, le taux de clics moyen est nettement supérieur à celui constaté lors de nos expérimentations sur la collection *OverBlog* (0,0990 contre 0,0389). Nous notons cependant que notre modèle conduit là encore à de meilleurs taux de clics. Il convient donc d'évaluer dans quelle mesure le domaine, lorsqu'il est très spécifique, peut impacter les performances du modèle.

De notre point de vue, cette thèse constitue une première brique pour des systèmes de recommandation basés sur l'apprentissage actif. En effet, le fait de diversifier la provenance des recommandations et d'apprendre comment combiner au mieux des éléments, implique que, à terme, les utilisateurs ont une influence sur le choix des exemples d'apprentissage. L'apprentissage actif est un domaine très dynamique et qui commence à intéresser la communauté

de la recherche d'information (Candillier et Lemaire, 2012). Certaines approches de ce type (Rubens *et al.*, 2011) pourraient être intégrées dans les systèmes de recommandation.

Publications de l'auteur

Articles de revues internationales

L. CANDILLIER, M. CHEVALIER, D. DUDOGNON et J. MOTHE : Multiple Similarities for Diversity in Recommender Systems, *International Journal On Advances in Intelligent Systems*, International Academy, Research and Industry Association (IARIA), Vol. 5 n° 3 & 4, pages 234-246, 2012

Conférences et workshops internationaux

M. CHEVALIER, T. DKAKI, D. DUDOGNON et J. MOTHE : Recommender system based on random walks and text retrieval approaches, *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases - Discovery Challenge Workshop (ECML/PKDD - DCW 2011)*, , Rudjer Boskovic Institute, pages 95-102, 2011

L. CANDILLIER, M. CHEVALIER, D. DUDOGNON et J. MOTHE : Diversity in Recommender Systems : Bridging the gap between users and systems, *International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC 2011)*, IARIA, pages 48-58, 2011 (Awarded paper)

D. DUDOGNON, G. HUBERT, J. MARCO, J. MOTHE, B. J. V. RALALASON, J. THOMAS, A. REYMONET, H. MAUREL, M. MBARKI, P. LAUBLET et V. ROUX, DYNAMic Ontology for Information Retrieval, *International Conference on Adaptivity, Personalization and Fusion of Heterogeneous Information (RIAO 2010)*, Centre de hautes études internationales d'Informatique Documentaire (C.I.D.), pages 213-215, 2010

Conférences et workshops nationaux

L. CANDILLER, M. CHEVALIER, D. DUDOGNON et J. MOTHE : Diversité de recommandations : application à une plateforme de blogs et évaluation, *Conférence francophone en Recherche d'Information et Applications (CORIA 2013)*, pages 269-276, 2013

D. DUDOGNON, G. HUBERT et B. J. V. RALALASON : ProxiGénéa : Une mesure de similarité conceptuelle,*Colloque Veille Stratégique Scientifique et Technologique (VSST 2010)*, Université Paul Sabatier, 2010

Conférences sans actes publiés

D. DUDOGNON : Plate-forme de recommandation de contenu multi-facette pour le Web 2.0, *Séminaire FREMIT*, 2010

Rapports

V. CAMPS, D. DUDOGNON, G. HUBERT, M. MBARKI, J. MOTHE, B. J. V. RALALASON, A. REYMONET, V. ROUX, Z. SELLAMI et J. THOMAS : DYNAMO (DYNAMic Ontology for information retrieval) : Validation, Evaluation du modèle - Livrable lot 9 (version 1), Rapport de contrat, Dynamo 2.1, IRIT, 2011

D. DUDOGNON : Compte rendu d'évaluation - Moteur de recherche de la plateforme OverBlog, Rapport de recherche, IRIT/RR–2012-13–FR, IRIT, 2010

Bibliographie

G. Adomavicius et Y.O. Kwon : Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, 24(5):896–911, 2012.

G. Adomavicius et A. Tuzhilin : Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.

N. Agarwal et H. Liu : Blogosphere : research issues, tools, and applications. *ACM SIGKDD Explorations Newsletter*, 10(1):18–31, 2008.

R. Agrawal, S. Gollapudi, A. Halverson et S. Ieong : Diversifying search results. *In 2nd ACM International Conference on Web Search and Data Mining*, pages 5–14. ACM, 2009.

M. Balabanović et Y. Shoham : Fab : content-based, collaborative recommendation. *Communications of the ACM*, 40(3):66–72, 1997.

C. Basu, H. Hirsh et W. Cohen : Recommendation as classification : Using social and content-based information in recommendation. *In AAAI/IAAI*, pages 714–720, 1998.

W. Bi, X. Yu, Y. Liu, F. Guan, Z. Peng, H. Xu et X. Cheng : Ictnet at web track 2009 diversity task. Rapport technique, DTIC Document, 2009.

P. Borlund : The concept of relevance in ir. *Journal of the American Society for Information Science and Technology (JASIST)*, 54(10):913–925, 2003.

M. Boughanem : *Les systèmes de recherche d'informations : d'un modèle classique à un modèle connexionniste*. Thèse de doctorat, Université de Toulouse, 1992.

K. Bradley et B. Smyth : Improving recommendation diversity. *In 12th National Conference in Artificial Intelligence and Cognitive Science (AICS-01)*, pages 75–84. Citeseer, 2001.

J. S. Breese, D. Heckerman et C. Kadie : Empirical analysis of predictive algorithms for collaborative filtering. *In 14th Conference on Uncertainty in artificial intelligence*, UAI'98, pages 43–52. Morgan Kaufmann Publishers Inc., 1998.

J. D. Brutlag, H. Hutchinson et M. Stone : User preference and search engine latency. *In JSM Proceedings, Qualtiy and Productivity Research Section.*, 2008.

R. Burke : Hybrid recommender systems : Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002.

R. Burke : Hybrid web recommender systems. *In The adaptive web*, pages 377–408. Springer, 2007.

G. Cabanac, M. Chevalier, C. Chrisment et C. Julien : An original usage-based metrics for building a unified view of corporate documents. *In Database and Expert Systems Applications*, pages 202–212. Springer, 2007.

L. Candillier, M. Chevalier, D. Dudognon et J. Mothe : Multiple similarities for diversity in recommender systems. *International Journal On Advances in Intelligent Systems*, 5(3 and 4):234–246, 2012.

L. Candillier, K. Jack, F. Fessant et F. Meyer : State-of-the-Art recommender systems. *In Collaborative and Social Information Retrieval and Access : Techniques for Improved User Modeling*, pages 1–22. IGI Global, 2009.

L. Candillier et V. Lemaire : Design and analysis of the nomao challenge - active learning in the real-world. *In Proceedings of the ALRA : Active Learning in Real-world Applications, Workshop ECML-PKDD*, 2012.

J. Carbonell et J. Goldstein : The use of mmr, diversity-based reranking for reordering documents and producing summaries. *In 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336. ACM, 1998.

P. Chandar, A. Kailasam, D. Muppaneni, L. Thota et B. Carterette : Ad hoc and diversity retrieval at the university of delaware. *In Text REtrieval Conf*, 2009.

O. Chapelle et Y. Zhang : A dynamic bayesian network click model for web search ranking. *In 18th international conference on World Wide Web*, WWW '09, pages 1–10. ACM, 2009.

M. Chevalier, T. Dkaki, D. Dudognon et J. Mothe : Recommender system based on random walks and text retrieval approaches. *In* T. Smuc, N. Antulov-Fantulin et M. Morzy, éditeurs : *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases - Discovery Challenge Workshop (ECML/PKDD - DCW)*, pages 95–102, http ://www.irb.hr/, 2011. Rudjer Boskovic Institute.

C.L. Clarke, N. Craswell et I. Soboroff : Overview of the trec 2009 web track. Rapport technique, DTIC Document, 2009.

C.L.A. Clarke, G.V. Cormack et E.A. Tudhope : Relevance ranking for one to three term queries. *Information processing & management*, 36(2):291–311, 2000.

C.L.A. Clarke, M. Kolla, G.V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher et I. MacKinnon : Novelty and diversity in information retrieval evaluation. *In 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 659–666, 2008.

M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes et M. Sartin : Combining content-based and collaborative filters in an online newspaper. *In Proceedings of ACM SIGIR workshop on recommender systems*, volume 60, 1999.

C. Cleverdon et M. Kean : Factors determining the performance of indexing systems. Aslib Cranfield Research Project, Cranfield, England, 1968.

N. Craswell, O. Zoeter, M. Taylor et B. Ramsey : An experimental comparison of click position-bias models. *In International Conference on Web Search and Data Mining*, pages 87–94. ACM, 2008.

D. Dudognon : Compte rendu d'évaluation - Moteur de recherche de la plateforme OverBlog. Rapport de recherche IRIT/RR–2012-13–FR, IRIT, Université Paul Sabatier, Toulouse, 2010.

D. Dudognon, G. Hubert, J. Marco, J. Mothe, B. Ralalason, J. Thomas, A. Reymonet, H. Maurel, M. Mbarki, P. Laublet et V. Roux : Dynamic ontology for information retrieval. *In Adaptivity, Personalization and Fusion of Heterogeneous Information*, pages 213–215. CID, 2010a.

D. Dudognon, G. Hubert et B. J. V. Ralalason : Proxigénéa : Une mesure de similarité conceptuelle. *In Colloque Veille Stratégique Scientifique et Technologique (VSST)*. Université Paul Sabatier, 2010b.

S. T Dumais, G. W. Furnas, T. K. Landauer, S. Deerwester et R. Harshman : Using latent semantic analysis to improve access to textual information. *In Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 281–285. ACM, 1988.

I. Esslimani, A. Brun et A. Boyer : A collaborative filtering approach combining clustering and navigational based correlations. *Web Information Systems and Technologies*, pages 364–369, 2009.

E. Fox et J. Shaw : Combination of multiple searches. *NIST Special Publication*, pages 243–243, 1994.

F. Guo, C. Liu et Y. M. Wang : Efficient multiple-click models in web search. *In Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 124–131, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-390-7.

D. Harman : Overview of the first text retrieval conference (trec-1). *In TREC*, pages 1–20, 1993.

D. Harman : An overview of the third text retrieval conference, national institute of standards and tehnology. *NIST Special Publication*, pages 500–225, 1994.

C. Hayes, P. Massa, P. Avesani et P. Cunningham : An on-line evaluation framework for recommender systems. *In Workshop on Personalization and Recommendation in E-Commerce*. Springer Verlag, 2002.

J. He, K. Balog, K. Hofmann, E. Meij, M. Rijke, M. Tsagkias et W. Weerkamp : Heuristic ranking and diversification of web documents. Rapport technique, DTIC Document, 2009.

M.A. Hearst, M. Hurst et S.T. Dumais : What should blog search look like ? *In Proceedings of the 2008 ACM Workshop on Search in social media*, pages 95–98. ACM, 2008.

J.L. Herlocker, J.A. Konstan, L.G. Terveen et J.T. Riedl : Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.

G. Hubert, Y. Loiseau et J. Mothe : Etude de différentes fonctions de fusion de systèmes de recherche d'information. *In CIDE 10 : Le document numérique dans le monde de la science et de la recherche*, pages 199–207. EUROPIA, 2007.

L.B. Jabeur, L. Tamine et M. Boughanem : A social model for literature access : Towards a weighted social network of authors. *In Adaptivity, Personalization and Fusion of Heterogeneous Information*, pages 32–39. CID, 2010.

M. Jahrer, A. Töscher et R. Legenstein : Combining predictions for accurate recommender systems. *In 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 693–702. ACM, 2010.

K. Järvelin et J. Kekäläinen : Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

T. Joachims, D. Freitag et T. Mitchell : Webwatcher : A tour guide for the world wide web. *In International Joint Conference on Artificial Intelligence (IJCAI)*, pages 770–777. Morgan Kaufmann, 1997.

T. Joachims, L. Granka, B. Pan, H. Hembrooke et G. Gay : Accurately interpreting clickthrough data as implicit feedback. *In 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 154–161. ACM, 2005.

R. Kaptein, M. Koolen et J. Kamps : Result diversity and entity ranking experiments : Anchors, links, text and wikipedia. Rapport technique, DTIC Document, 2009.

P. Kolari, A. Java, T. Finin, J. Mayfield, A. Joshi et J. Martineau : Blog track open task : Spam blog classification. *TREC Blog Track Notebook*, 2006.

J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R Gordon et J. Riedl : Grouplens : applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87, 1997.

Y. Koren, R. Bell et C. Volinsky : Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

A. Kritikopoulos, M. Sideri et I. Varlamis : Blogrank : ranking weblogs based on connectivity and similarity features. *In 2nd international workshop on Advanced architectures and algorithms for internet delivery and applications*, page 8. ACM, 2006.

R. Kumar, J. Novak, P. Raghavan et A. Tomkins : On the bursty evolution of blogspace. *World Wide Web*, 8(2):159–178, 2005.

F.W. Lancaster : *Information retrieval systems ; characteristics, testing and evaluation*. J. Wiley, 1968.

L. Laporte, L. Candillier, S. Déjean et J. Mothe : Évaluation de la pertinence dans les moteurs de recherche géoréférencés. *In INFORSID*, pages 281–298, 2012.

J.H. Lee : Analyses of multiple evidence combination. *In ACM SIGIR Forum*, volume 31, pages 267–276. ACM, 1997.

S.T. Li, C.C. Chen et F. Huang : Conceptual-driven classification for coding advise in health insurance reimbursement. *Artificial Intelligence in Medicine*, 51(1):27–41, 2011.

H. Lieberman : Letizia : an agent that assists web browsing. *In 14th international joint conference on Artificial intelligence - Volume 1*, IJCAI'95, pages 924–929. Morgan Kaufmann Publishers Inc., 1995.

R. Likert : A technique for the measurement of attitudes. *Archives of psychology*, 1932.

D. Lin : An information-theoretic definition of similarity. *In ICML*, volume 98, pages 296–304, 1998.

J. Lin, D. Metzler, T. Elsayed et L. Wang : Of ivory and smurfs : Loxodontan mapreduce experiments for web search. Rapport technique, DTIC Document, 2009.

G. Linden, B. Smith et J. York : Amazon. com recommendations : Item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.

H.P. Luhn : The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165, 1958.

J. MacQueen : Some methods for classification and analysis of multivariate observations. *In 5th Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

B. Magnini et C. Strapparava : Improving user modelling with content-based techniques. *In 8th International Conference on User Modeling*, pages 74–83. Springer, 2001.

T. W. Malone, K. R. Grant, F. A. Turbak, S. A. Brobst et M. D. Cohen : Intelligent information-sharing systems. *Commun. ACM*, 30(5):390–402, 1987.

R. McCreadie, C. Macdonald, I. Ounis, J. Peng et R.L. Santos : University of glasgow at trec 2009 : Experiments with terrier. Rapport technique, DTIC Document, 2009.

S. M. McNee, J. Riedl et J. A. Konstan : Being accurate is not enough : how accuracy metrics have hurt recommender systems. *In CHI '06 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '06, pages 1097–1101, New York, NY, USA, 2006. ACM. ISBN 1-59593-298-4.

P. Melville, R. J. Mooney et R. Nagarajan : Content-boosted collaborative filtering for improved recommendations. *In AAAI/IAAI*, pages 187–192, 2002.

G. Mishne : Information access challenges in the blogspace. *In the International Workshop on Intelligent Information Access (IIIA 2006)*. Citeseer, 2006.

S. Mizzaro : How many relevances in information retrieval ? *Interacting with Computers*, 10(3):303–320, 1998.

M. Montaner, B. Lòpez et J. L. de la Rosa : A taxonomy of recommender agents on the internet. *Artificial Intelligence Review*, 19(4):285–330, 2003.

J. Mothe : Search mechanisms using a new neural network model comparison with the vector space model. *In* Jean-Louis Funck-Brentano et Frederick Seitz, éditeurs : *RIAO*, pages 275–295. CID, 1994.

J. Mothe, C. Chrisment, T. Dkaki, B. Dousset et S. Karouach : Combining mining and visualization tools to discover the geographic structure of a domain. *Computers, environment and urban systems*, 30(4):460–484, 2006.

R. Picot-Clémente : *Une architecture générique de Systèmes de recommandation de combinaison d'items : application au domaine du tourisme.* Thèse de doctorat, Université de Bourgogne, 2011.

K. Pinel-Sauvagnat et J. Mothe : Mesures de la qualité des systèmes de recherche d'information. *Ingénierie des Systèmes d'Information*, 18(3):11–38, 2013.

J. M. Ponte et W. B. Croft : A language modeling approach to information retrieval. *In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM, 1998.

M.F. Porter : An algorithm for suffix stripping. *Program : electronic library and information systems*, 14(3):130–137, 1980.

J. R. Quinlan : *C4. 5 : programs for machine learning*, volume 1. Morgan kaufmann, 1993.

F. Radlinski, P.N. Bennett, B. Carterette et T. Joachims : Redundancy, diversity and interdependent document relevance. *In ACM SIGIR Forum*, volume 43, pages 46–52. ACM, 2009.

P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom et J. Riedl : Grouplens : an open architecture for collaborative filtering of netnews. *In ACM conference on Computer supported cooperative work*, CSCW '94, pages 175–186. ACM, 1994.

F. Ricci, L. Rokach et B. Shapira : Introduction to recommender systems handbook. *In* Francesco Ricci, Lior Rokach, Bracha Shapira et Paul B. Kantor, éditeurs : *Recommender Systems Handbook*, pages 1–35. Springer US, 2011.

S. E Robertson et K. Spärck Jones : Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3):129–146, 1976.

S.E. Robertson, S. Walker, S. Jones, M.M. Hancock-Beaulieu, M. Gatford *et al.* : Okapi at trec-3. *NIST SPECIAL PUBLICATION SP*, pages 109–109, 1995.

N. Rubens, D. Kaplan et M. Sugiyama : Active learning in recommender systems. *In Recommender Systems Handbook*, pages 735–767. Springer, 2011.

G. Salton : The smart retrieval system experiments in automatic document processing. Prentice-Hall, Inc., 1971.

G. Salton, E. A. Fox et H. Wu : Extended boolean information retrieval. *Communications of the ACM*, 26(11):1022–1036, 1983.

G. Salton et M.J. McGill : *Introduction to modern information retrieval*. McGraw-Hill, Inc., 1983.

R.L.T. Santos, C. Macdonald et I. Ounis : Selectively diversifying web search results. *In 19th ACM international conference on Information and knowledge management*, pages 1179–1188. ACM, 2010.

B. Sarwar, G. Karypis, J. Konstan et J. Riedl : Analysis of recommendation algorithms for e-commerce. *In Proceedings of the 2nd ACM conference on Electronic commerce*, pages 158–167. ACM, 2000a.

B. M. Sarwar, G. Karypis, J. A. Konstan et J. T. Riedl : Application of dimensionality reduction in recommender systems - a case study. *In ACM WebKDD Workshop*, 2000b.

J. Schafer, D. Frankowski, J. Herlocker et S. Sen : Collaborative filtering recommender systems. *The adaptive web*, pages 291–324, 2007.

J.B. Schafer, J.A. Konstan et J. Riedl : Meta-recommendation systems : user-controlled integration of diverse recommendations. *In 11th international conference on Information and knowledge management*, pages 43–51. ACM, 2002.

G. Semeraro, M. Degemmis, P. Lops et P. Basile : Combining learning and word sense disambiguation for intelligent user profiling. *In IJCAI*, volume 7, pages 2856–2861, 2007.

B. Sheth et P. Maes : Evolving agents for personalized information filtering. *In 9th IEEE Conference on Artificial Intelligence for Applications*, 1993.

M. Siegler, M. J. Witbrock, S. Slattery, K. Seymore, R. E. Jones et A. G. Hauptmann : Experiments in spoken document retrieval at cmu. *In TREC*, pages 291–302, 1997.

K. Spärck Jones : A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

K. Spärck Jones : Document retrieval : Shallow data, deep theories ; historical reflections, potential directions. *Lecture Notes in Computer Science*, 2633:1–11, 2003.

K. Spärck-Jones, S. E. Robertson et M. Sanderson : Ambiguous requests : implications for retrieval tests, systems and theories. *In ACM SIGIR Forum*, volume 41, pages 8–17. ACM, 2007.

M.A. Tayebi, S.M. Hashemi et A. Mohades : B2rank : An algorithm for ranking blogs based on behavioral features. *In Web Intelligence, IEEE/WIC/ACM International Conference on*, pages 104–107. IEEE, 2007.

E. G. Toms : Serendipitous information retrieval. *In DELOS Workshop : Information Seeking, Searching and Querying in Digital Libraries*. Zurich, 2000.

S. Vargas et P. Castells : Rank and relevance in novelty and diversity metrics for recommender systems. *In 5th ACM Conference on Recommender Systems (RecSys 2011)*, pages 109–116, 2011.

C.C. Vogt et G.W. Cottrell : Fusion via a linear combination of scores. *Information Retrieval*, 1(3):151–173, 1999.

L. Von Ahn, M. Blum, N. Hopper et J. Langford : "captcha : Using hard ai problems for security". *Advances in Cryptology EUROCRYPT 2003*, pages 646–646, 2003.

J. Wang, A. P. de Vries et M. J. T. Reinders : Unifying user-based and item-based collaborative filtering approaches by similarity fusion. *In 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 501–508. ACM, 2006.

Z. Wu et M. Palmer : Verbs semantics and lexical selection. *In 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics, 1994.

Y. Xu et H. Yin : Novelty and topicality in interactive information retrieval. *Journal of the American Society for Information Science and Technology*, 59 (2):201–215, 2008.

Y.C. Xu et Z. Chen : Relevance judgment : What do information users consider beyond topicality ? *Journal of the American Society for Information Science and Technology*, 57(7):961–973, 2006.

W. Zheng et H. Fang : Axiomatic approaches to information retrieval-university of delaware at trec 2009 million query and web tracks. Rapport technique, DTIC Document, 2009.

C.N. Ziegler, S.M. McNee, J.A. Konstan et G. Lausen : Improving recommendation lists through topic diversification. *In 14th international conference on World Wide Web*, pages 22–32. ACM, 2005.

G.K. Zipf : *Human behavior and the principle of least effort : an introduction to human ecology.* Addison-Wesley Press, 1949.

Diversity and recommender system : application to a high traffic blog platform

Recommender Systems (RS) aim at automatically providing objects related to user's interests. Considering content, user's interests can be modeled from the visited content and user's actions. However, modeling is complex and recommendations are often far away from the real user's interests.

To obtain more relevant recommendations for each user, we propose a SR model that builds a list of recommendations aiming at covering a large range of interests, even when only a few information about the user is available. The SR model we propose is based on diversity. It uses different interest measures and an aggregation function to build the final set of recommendations.

We demonstrate the interest of our approach, and finally, we evaluate our model on the OverBlog platform to validate its scalability in an industrial context.

Keywords : recommender systems, diversity, interest measure, content platform, blog, cold start

Thèse de doctorat soutenue par Damien DUDOGNON
à l'Université de Paul Sabatier le 04/04/2014
sous la direction de Josiane MOTHE et Max CHEVALIER

École Doctoral MITT - Spécialité Image, Information, Hypermedia

Diversité et système de recommandation : application à une plateforme de blogs à fort trafic

Les systèmes de recommandations (SR) ont pour objectif de proposer automatiquement aux usagers des objets en relation avec leurs intérêts. Dans le contexte des plateformes de contenus, ces intérêts peuvent être modélisés à partir des contenus des documents visités ou des actions réalisées. Cette modélisation est complexe, conduisant à des recommandations souvent éloignées des intérêts réels.

Pour tendre vers des recommandations plus pertinentes, nous proposons un SR qui construit une liste de recommandations répondant à un large spectre d'intérêts potentiels, et ce même lorsque le système ne possède que peu d'information sur l'usager. L'originalité de notre modèle est qu'il repose sur la notion de diversité. Elle est obtenue en agrégeant différentes mesures d'intérêts pour construire la liste finale.

Après avoir démontré l'intérêt de notre approche, nous évaluons notre modèle sur la plateforme de blogs OverBlog pour le valider dans un contexte industriel et à grande échelle.

Mots clés : système de recommandation, diversité, mesure de sélection, plateforme de contenus, blog, démarrage à froid

Institut de Recherche en Informatique de Toulouse (UMR 5505)
IRIT - Université Toulouse 3 Paul Sabatier
118 Route de Narbonne
F-31062 TOULOUSE CEDEX 9